

SnapForm

Handbuch Designer 5.x

1	**Einleitung**	**1 - 1**
1.1	Willkommen in der Welt von Snapform!	1
1.2	Zu diesem Handbuch	2
1.2.1	Zielgruppe	2
1.2.2	Konventionen	2
1.2.2.1	Darstellungen im Text	2
1.2.2.2	Screenshots	3
1.2.2.3	Anmerkungen	4
1.2.3	Beschriebene Softwareversionen	4
1.3	Komponenten von Snapform	5
1.3.1	Snapform PDF Converter	5
1.3.2	Snapform Designer	5
1.3.3	Snapform Viewer	5
1.3.4	Workflow- und Management-Tools	6
1.3.5	API und SDK	6
1.4	Das QDF-Dokumentenformat	7
1.4.1	Allgemeines	7
1.4.2	QDF/S	8
1.4.3	QDF/A	9
2	**Installation**	**2 - 1**
2.1	Systemvoraussetzungen	1
2.2	Vorbereitung des Systems	2
2.3	Standardinstallation	2
2.3.1	Installation von Snapform Designer	3
2.3.2	Installation von Snapform Viewer	3
2.3.3	Installation von Snapform PDF Converter	4
2.4	Registrierung	4
2.5	Konfiguration	6
2.6	Deinstallation	6
3	**Verwendung des Designers**	**3 - 1**
3.1	Programm starten und beenden	1
3.1.1	Programmstart	1
3.1.2	Programm beenden	2
3.2	Komponenten von Snapform Designer	3
3.2.1	Das Programmfenster	3
3.2.2	Menuleiste	4
3.2.2.1	Das Apfel-Menu (Macintosh-spezifisch)	4
3.2.2.2	Das Menu Snapform Designer (Macintosh-spezifisch)	4
3.2.2.3	Das Menu Datei	7
3.2.2.4	Das Menu Bearbeiten	10

3.2.2.5	Das Menu Format	13
3.2.2.6	Das Menu Steuerelemente	15
3.2.2.7	Das Menu Ansicht	17
3.2.2.8	Das Menu Extras	18
3.2.2.9	Das Menu Hilfe	19
3.2.3	Werkzeugleiste	20
3.2.4	Steuerelemente-Leiste	24
3.2.5	Arbeitsfläche/Editor-Fenster	24
3.2.6	Eigenschaftenfenster	25
3.2.7	Extras-Fenster	26
3.2.7.1	Objekte	26
3.2.7.2	Werkzeuge	27
3.2.7.3	Suchen und ersetzen	27
3.2.8	Formeleditor	29
3.2.9	Weitere Komponenten der Benutzeroberfläche	31
3.2.9.1	Tastaturkürzel	31
3.2.9.2	Farbauswahlfenster	31
3.3	Betriebsarten von Designer	35
3.3.1	Entwurfs-Ansicht	35
3.3.2	Vorschau	35
4	**Formular-Entwicklung**	**4 - 1**
4.1	Einleitung	1
4.2	Analyse und Planung	2
4.3	Grundlayout	4
4.3.1	Hintergrundbilder	4
4.3.1.1	Snapform PDF Converter	4
4.3.1.2	Konversions-Service	5
4.3.2	Der Snapform PDF Converter	5
4.3.2.1	Vorbereitung der PDF-Dokumente	5
4.3.2.2	Die Benutzeroberfläche des Snapform PDF Converters	7
4.3.2.3	Grundeinstellungen	12
4.3.2.4	Einzeldokument konvertieren	14
4.3.2.5	Batch-Konversion	14
4.3.2.6	Probleme bei der Konversion	16
4.3.3	Neues Dokument erstellen	17
4.3.3.1	Import bestehender Hintergrundseiten	18
4.3.3.2	Leere Seiten erzeugen	19
4.3.3.3	Eigenschaften von neuen Formularen	21
4.3.4	Dokumentseiten	23
4.3.4.1	Seiten hinzufügen und entfernen	23
4.3.4.2	Seiteneigenschaften	25

4.3.5	Grafikelemente	29
4.3.5.1	Linien	29
4.3.5.2	Rechtecke	36
4.3.5.3	Flächen	42
4.3.5.4	Bilder und Logos	48
4.3.6	Text	58
4.3.6.1	Das Beschriftungsfeld-Werkzeug	59
4.3.6.2	In Snapform verwendbare HTML-Tags und Attribute	70
4.4	Formularebene	71
4.4.1	Labels und Formelcontainer	71
4.4.2	Eingabefelder	72
4.4.2.1	Eingabefelder im Allgemeinen	72
4.4.2.2	Textfelder	80
4.4.2.3	Mehrzeilige Felder	84
4.4.2.4	Numerische Felder	87
4.4.2.5	Datumsfelder	96
4.4.2.6	Eingabemasken	101
4.4.2.7	Passwortfelder	107
4.4.2.8	Eingabefelder-Beispiele	111
4.4.3	Kontrollkästchen und Optionsfelder	114
4.4.3.1	Kontrollkästchen	114
4.4.3.2	Optionsfelder	120
4.4.3.3	Optionsfelder aufbauen	124
4.4.3.4	Beispiele von Kontrollkästchen und Optionsfeldern	127
4.4.4	Auswahllisten	128
4.4.5	1D-Barcodes	137
4.4.5.1	Das Barcodewerkzeug	138
4.4.5.2	Unterstützte 1D-Barcodes	149
4.4.5.3	1D-Barcode-Beispiele	155
4.4.6	2D-Barcodes	162
4.4.6.1	Allgemeines zu 2D-Barcodes	163
4.4.6.2	Aztec Barcode	164
4.4.6.3	PDF417 Barcode	176
4.4.6.4	Datamatrix	187
4.4.6.5	QRCode	198
4.4.7	Tabellen	209
4.4.7.1	Das Tabellenwerkzeug	210
4.4.7.2	Aufbau von Tabellen	221
4.4.7.3	Tabellen-Beispiele	233
4.4.8	Aktive Elemente	237
4.4.8.1	Hyperlinks	237

4.4.8.2 Hinweise/Infopunkte 245
4.4.8.3 Schaltflächen 258
4.4.8.4 Angewandte Aktive Elemente: ein Hilfesystem 259
4.5 Aktionen und Logik 262
4.5.1 Träger von Aktionen und Logik 262
4.5.1.1 Felder 263
4.5.1.2 Funktionen 263
4.5.2 In Snapform verfügbare Aktionen und Logik 264
4.5.2.1 String-Funktionen 265
4.5.2.2 Regular Expression-Funktionen 266
4.5.2.3 Mathematik-Funktionen 267
4.5.2.4 Tabellen-Funktionen 269
4.5.2.5 Datums-Funktionen 270
4.5.2.6 Status-Funktionen 272
4.5.2.7 Konsistenzprüfungs-Funktionen 273
4.5.2.8 Identifikations-Funktionen 274
4.5.2.9 Combobox-Funktionen 275
4.5.2.10 Spezial-Funktionen 276
4.6 Integrität des Formulars 278
4.7 Stilvorlagen 280
4.7.1 Der Stilvorlageneditor 280
4.7.2 Stilvorlagen importieren und exportieren 286
4.8 Formular- und Sicherheitseinstellungen 287
4.8.1 Einleitung 287
4.8.2 Allgemeine Einstellungen 288
4.8.2.1 Dokument-Angaben 288
4.8.2.2 Vorlagen 289
4.8.2.3 Druck 290
4.8.2.4 Dateiformat 290
4.8.3 Sicherheit 291
4.8.3.1 Passwörter 292
4.8.3.2 Optionen 293
4.8.3.3 Befehlstasten 295
4.8.4 Prüfung 298
4.8.4.1 Formular mit Ablaufdatum 298
4.8.4.2 Ungültige Formulare 299
4.8.4.3 Unvollständige Formulare 300
4.8.5 RDF-Metadaten 300
4.8.6 Serverseitige Interaktion 301
4.8.7 Schriftarten 303
4.8.8 eFiling (Datenübertragung) 304

4.8.8.1 Format 306
4.8.8.2 Dateiname 307
4.8.8.3 Ziel 307
4.8.8.4 UI Einstellungen 308
4.9 Freigabe und Veröffentlichung 309
4.10 Formularmanagement 311

5 Formelsprache-Referenz 5 - 1
5.1 Konstanten 3
5.1.1 TRUE 3
5.1.2 FALSE 3
5.1.3 pi 3
5.1.4 e 4
5.1.5 null 4
5.2 Operatoren 5
5.2.1 x + y 5
5.2.2 x - y 5
5.2.3 x * y 6
5.2.4 x / y 6
5.2.5 x ^ y 6
5.2.6 x % y 7
5.2.7 x = y 7
5.2.8 x <> y 7
5.2.9 x >= y 8
5.2.10 x > y 8
5.2.11 x <= y 9
5.2.12 x < y 9
5.2.13 x AND y 9
5.2.14 x OR y 10
5.2.15 NOT x 10
5.3 Funktionen 11
5.3.1 Abs(x) 11
5.3.2 Acos(x) 11
5.3.3 addDay(date_value, number) 12
5.3.4 addHour(time_value, number) 13
5.3.5 addMinute(time_value, number) 14
5.3.6 addMonth(date_value, number) 15
5.3.7 addYear(date_value, number) 16
5.3.8 Asin(x) 17
5.3.9 Atan(x) 18
5.3.10 calcDays(date_first, date_second) 18
5.3.11 calcHours(date_first, date_second) 19

5.3.12	calcMinutes(date_first, date_second)	20
5.3.13	Ceil(x)	21
5.3.14	Cell(table, column, row)	22
5.3.15	checkCardNumber(value)	23
5.3.16	checkMod10(value)	24
5.3.17	checkMod10r(value)	24
5.3.18	Compress(x)	25
5.3.19	Cos(x)	26
5.3.20	Cover(x)	26
5.3.21	createGUID()	27
5.3.22	Date(year, month, day)	28
5.3.23	Exp(x)	29
5.3.24	find(string, regex[, start])	29
5.3.25	Floor(x)	30
5.3.26	getBytes(string)	31
5.3.27	getCol()	32
5.3.28	getContent(url)	32
5.3.29	getDataID()	33
5.3.30	getDay(date_value)	34
5.3.31	getHour(time_value)	34
5.3.32	getMinute(time_value)	35
5.3.33	getMod10(value)	36
5.3.34	getMod10r(value)	36
5.3.35	getMonth(date_value)	37
5.3.36	getPrintID()	38
5.3.37	getPrintIdStr(length [, digits]	38
5.3.38	getRandomString(length [, digits])	39
5.3.39	getRow()	40
5.3.40	getSelectedIndex(ComboBox)	41
5.3.41	getSessionID()	41
5.3.42	getValues(delimiter, value1 [, value2[, …]])	42
5.3.43	getVersion()	43
5.3.44	getYear(date_value)	44
5.3.45	if(logical_test, value_if_true [,value_if_false](	44
5.3.46	importData(url)	45
5.3.47	IncVersion(initial_value, step)	46
5.3.48	InStr(string, pattern [, start_position])	47
5.3.49	isEmpty(x)	48
5.3.50	Left(string, count)	49
5.3.51	Len(x)	50
5.3.52	Log(x)	50

5.3.53 match(string, regex) 51
5.3.54 Max(x, y) 52
5.3.55 Min(x, y) 52
5.3.56 Now() 53
5.3.57 replaceAll(orig_string, regex, replace_with) 54
5.3.58 Right(string, count) 54
5.3.59 Round(x) 55
5.3.60 setFieldPrintable(true|false, field_name) 56
5.3.61 setFieldVisible(true|false, field_name) 57
5.3.62 setFieldWritable(true|false, field_name) 57
5.3.63 setList(ComboBox, str_items, str_values) 58
5.3.64 setPagePrintable(true|false, page_number [, page_number[,…]]) 59
5.3.65 setPageVisible(true|false, page_number [, page_number[, …]]) 60
5.3.66 setPrintable(category_list, true|false) 61
5.3.67 setSelectedIndex(ComboBox, index) 62
5.3.68 setVisible(category_list, true|false) 62
5.3.69 setWritable(category_list, true|false) 63
5.3.70 Sin(x) 64
5.3.71 Sqrt(x) 65
5.3.72 Str(x) 65
5.3.73 StrTotal(table, column) 66
5.3.74 SubArray(byte_array, start [, count]) 66
5.3.75 SubStr(string, start[, count]) 67
5.3.76 Tan(x) 68
5.3.77 TblValues(table, tbl_start, tbl_end, row_start, row_end, delimiter, column [, column[, …]]) 69
5.3.78 Time(hour, minute[, second]) 71
5.3.79 TimeTotal(table, column) 72
5.3.80 Today() 72
5.3.81 Total(table, column) 73
5.3.82 TrimStr(string) 74
5.3.83 Trunc(x) 74
5.4 Regular Expressions 76
5.4.1 Übersichtstabelle 76

6 Snapform-Dateiformate 6 - 1
6.1 Formular, komprimiert (QDF/S) 1
6.2 Formular, offen (QDF/A) 1
6.3 Formular-Template, komprimiert (QDF/S) 4
6.4 Formular-Template, offen (QDF/A) 4

6.5	Hintergrundebene	5
6.6	Feldebene	5
6.7	Daten, binär	5
6.8	Daten, offen	6
6.9	Das Snapform-Schema	6
7	**API und Toolkits**	**7 - 1**

1 Einleitung

1.1 Willkommen in der Welt von Snapform!

Snapform ist ein umfassendes System zur Entwicklung, Benutzung und Verwaltung von elektronischen Formularen.

Die Suite besteht aus verschiedenen Komponenten: Der Snapform Designer ist das Entwicklungswerkzeug, mit welchem die Formulare erstellt werden. Snapform Viewer ist die Client-Komponente, mit welcher die Formulare ausgefüllt und benutzt werden. Die Snapform Workflow- und Management-Tools dienen zum Verwalten von Formularen und Daten.

Snapform benutzt das Dateiformat QDF, welches auf XML-Technologien aufbaut. Die Daten liegen im XML-Format vor und können daher auf einfache Art und Weise weiterverarbeitet werden.

Snapform erlaubt neben einer elektronischen Variante auch die Integration in Papier-basierte Arbeitsabläufe. Barcodes vermindern hierbei Medienbrüche.

Der Formularentwickler wird durch eine einzigartige Palette von Eingabeobjekten und vordefinierten Formeln unterstützt. Ausserdem ist der Snapform Viewer in über 32 Sprachversionen verfügbar und damit geradezu prädestiniert für Formulare im globalen Umfeld.

1.2 Zu diesem Handbuch

1.2.1 Zielgruppe

Dieses Handbuch richtet sich an Forms Manager, Formularentwickler sowie Verantwortliche für elektronische Formulare, welche SnapForm-Formulare planen, erstellen, warten und verteilen. Zudem soll es Anwendern von Snapform-Formularen ausreichend Hintergrund-Informationen bieten, damit sie ihre Tätigkeit entsprechend erfolgreich ausüben können.

1.2.2 Konventionen

1.2.2.1 Darstellungen im Text

In diesem Handbuch gelten verschiedene Konventionen in der Darstellung.

Menubezeichnungen werden **in dieser Schriftart** dargestellt.

Schaltflächen werden entweder als Symbol oder mit ihrer Bezeichnung **in dieser Schriftart** dargestellt.

Texte von GUI-Elementen, wie Dialogfenster, Fehlermeldungen, etc. werden **in dieser Schriftart** dargestellt.

Tastenbezeichnungen werden wie folgt dargestellt: **<Ctrl><C>**. Dabei ist zu beachten, dass die auf der Tastatur aufgetragene Bezeichnung je nach Herkunftsland und Lokalisierung der betreffenden Tastatur abweichen kann.

Tastatureingaben werden **in dieser Schriftart dargestellt**.

Systemantworten im interaktiven Dialog werden **in dieser Schiftart dargestellt**.

Formelelemente (z.B. Funktionsbezeichnungen) oder Dateinamen werden im Text **`in dieser Schriftart dargestellt`**.

Platzhalter für Feldbezeichnungen und Variablen werden im Text und in Listings folgendermassen dargestellt: **`<Feldname>`**. Damit Formeln funktionieren, müssen diese Platzhalter durch real vorkommende Bezeichnungen ersetzt werden.

Formeln bzw. Programmcode wird als Listing **`in dieser Schrift dargestellt`**.

1.2.2.2 Screenshots

Screenshots dienen zur Orientierung auf dem Bildschirm, zur Darstellung von Elementen der Benutzeroberfläche, sowie zur visuellen Darstellung von Beispielen. Da Snapform Designer plattformunabhängig ist und für GUI-Elemente (Graphical User Interface; Grafische Benutzeroberfläche) auf Funktionen des jeweiligen Betriebssystem zugreift, kann die effektive Anzeige auf dem Bildschirm von den in diesem Handbuch gezeigten Screenshots abweichen. Für dieses Handbuch wurden Screenshots in einer Windows- und einer OS X-Umgebung gemacht, und wechselseitig eingesetzt.

In den Fällen, in welchen plattformspezifische Sachverhalte dargestellt werden, wird dies ausdrücklich erwähnt.

Hinweis: *In gewissen Fällen können Screenshots zur Verbesserung der Darstellung von Sachverhalten retuschiert worden sein.*

Es wird vorausgesetzt, dass die Anwender von SnapForm Designer mit der Benutzung ihres Rechners vertraut sind, und die Grundoperationen von Betriebssystem und normalen Anwendungen beherrschen. Bei den Schritt-für-Schritt-Anleitungen wird daher zu einem grossen Teil auf die Abbildung von Screenshots verzichtet.

1.2.2.3 Anmerkungen

Anmerkungen werden entsprechend der Wichtigkeit mit “Vorsicht”, “Hinweis” und “Praxistip” gekennzeichnet.

Vorsicht: *Eine mit “Vorsicht” gekennzeichneter Anmerkung enthält Informationen, welche wichtig sind und unbedingt beachtet werden müssen, da sie sonst zu schwerwiegenden Fehlfunktionen oder Datenverlust führen können.*

Hinweis: *Eine mit “Hinweis” gekennzeichnete Anmerkung enthält zusätzlche Informationen, welche zur Erklärung von Sachverhalten dienen, oder Ergänzungen zum Text enthalten. Es ist sehr sinnvoll, Hinweise zu beachten.*

Praxistip: *Eine mit “Praxistip” gekennzeichnete Anmerkung enthält Informationen, welche die Arbeit mit dem Programm erleichtern und weitgehend aus der Praxis stammen.*

1.2.3 Beschriebene Softwareversionen

Dieses Handbuch beschreibt das Verhalten von Snapform Designer Version 5.1.9 und berücksichtigt Snapform Viewer Version 1.5.24. Es gilt sinngemäss auch für ältere Versionen, wobei gewisse Funktionen eventuell nicht vorhanden sind.

1.3 Komponenten von Snapform

1.3.1 Snapform PDF Converter

Der Formatkonverter dient zum Konvertieren von PDF-Dokumenten in das Snapform-Dateiformat. Dadurch wird das erneute Layouten vorhandener Formularvorlagen hinfällig. Der Formularentwickler kann sich dank des PDF Converters schwergewichtig auf die Implementation der Formularlogik konzentrieren und erzielt dadurch erhebliche Zeiteinsparungen.

Diese Applikation ist nur für Windows erhältlich. Als Alternative zum Konverter bietet Ringler Inormatik AG einen Konvertierungsservice an.

Der Konverter ist im Abschnitt 4.3.2 genauer beschrieben.

1.3.2 Snapform Designer

Der Snapform Designer ist das Entwicklungswerkzeug für Snapform-Formulare. Diese Applikation basiert auf der Java Runtime Engine und ist für Windows sowie Macintosh ab OS X erhältlich.

Das vorliegende Handbuch beschreibt vorwiegend die Funktionalitäten im Snapform Designer.

1.3.3 Snapform Viewer

Der Snapform Viewer ist die Client-Komponente für Snapform-Formulare, und kann frei heruntergeladen werden. Er basiert auf der Java Runtime Engine und ist für Windows, Macintosh und verschiedene Unix/Linux Varianten erhältlich.

Der Snapform Viewer ist weitgehend selbsterklärend.

1.3.4 Workflow- und Management-Tools

Die Workflow- und Management-Tools erweitern Snapform zu einer vollständigen eForms-Lösung und ermöglichen ein zentrales Datenmanagement sowie eine zentral gesteuerte Sicherheitspolitik für die Formulare und deren Daten.

Die Workflow- und Management-Tools sind nicht im Umfang dieser Dokumentation beschrieben. Für weitere Informationen steht Ringler Informatik AG zur Verfügung.

1.3.5 API und SDK

Für Snapform-Anwendungen besteht ein API mit entsprechendem SDK. Diese Komponenten sind nicht Bestandteil dieser Dokumentation. Für weitere Informationen steht Ringler INformatik AG zur Verfügung.

1.4 Das QDF-Dokumentenformat

1.4.1 Allgemeines

Das Dokumentenformat für Snapform heisst QDF (Quick Document Format) und basiert auf XML. QDF besteht im Wesentlichen aus den folgenden fünf Komponenten:

- reduzierte Form von SVG zur Darstellung von Hintegrundlayout und Grafik
- Font-Ersatz- und -Einbettungsmechanismus
- Objekt-Ebene mit vektorbasierten Dateneingabe-Komponenten
- Daten-Ebene, welche Daten in einer strukturierten Art und Weise repräsentiert
- Strukturiertes Speichersystem zum Bündeln dieser Elemente sowie weiterer Anhänge in einer einzigen Datei, wobei Daten wahlweise bzw. soweit sinnvoll komprimiert werden

Die Gesamtheit dieser fünf Komponenten führt zu bedeutenden Vorteilen:

- QDF-Dokumente sind logisch vorformatiert und werden dadurch sehr kompakt
- Formularfelder und Business-Logiken sind ebenfalls sehr kompakt und vergrössern das Dokument nur in geringem Masse
- Die SVG-basierte Hintergrundebene des Formulars liefert Druckformate in Vektorqualität und garantiert vollständige Unterstützung von Transparenz
- Die Font-Ersatz-Strategie von Snapform ermöglicht eine Plattform übergreifende, naturgetreue Darstellung von Schriften, auch wenn sie auf dem Rechner des Anwenders nicht vorhanden sind.

Weitere Merkmale von QDF/Snapform sind:

- Exakte Wiedergabe des papierbasierten Formularlayouts am Bildschirm und beim Drucken erlaubt einen Einsatz auch in dokumentbasierten Arbeitsabläufe, bei welchen sowohl die Daten wie auch die Präsentation wichtig ist.
- Unterstützung aller gängigen Barcodes (1D und 2D) vermindern Medienbrüche und erlauben gleichzeitig ein Anbinden an bestehende Business-Applikationen.
- Die Unterstützung einer breiten Palette von Texteingabemethoden (einschliesslich Chinesisch, Japanisch, Koreanisch, Hebräisch und Arabisch) erlaubt einen weltweiten Einsatz von Formularen.
- Die Benutzeroberfläche des Snapform Viewers ist in über 32 Sprachen verfügbar und unterstützt dadurch Anwender in der breiten Öffentlichkeit.

Das Lizenzmodell von Snapform sieht die kostenlose und uneingeschränkte Nutzung von Snapform-Formularen. Für den Formularanwender bestehen damit keine Limitierungen, welche nur mit Zusatz-Software (oder Services) freigegeben werden.

Eine detailliertere Beschreibung des QDF-Dateiformats (soweit es ausserhalb von Spezialentwicklungen offen gelegt ist, befindet sich im Kapitel 6.

QDF kann verschlüsselt oder als offene XML-Datei formatiert werden:

1.4.2 QDF/S

QDF/S ist die am stärksten geschützte und gleichzeitig auch kompakteste Variante von QDF. Hier sind sämtliche Komponenten komprimiert und 128-bit verschlüsselt und können nur von Snapform Viewer und entsprechend ausgestalteten Server-Applikationen geöffnet werden.

1.4.3 QDF/A

QDF/A ist die "Standard"-Version von QDF, bei welcher die Daten-Ebene im XML-Format offen gelegt ist. Dadurch können ausgefüllte Snapform-Formulare von einem Archiv- bzw. Datenverwaltungssystem eingesehen und ohne Abhängigkeit von weiteren Tools einfach verwaltet werden.

2 Installation

2.1 Systemvoraussetzungen

Damit Snapform installiert und betrieben werden kann, müssen ein paar Systemanforderungen erfüllt sein.

Plattform

Für Snapform Viewer gilt grundsätzlich, dass alle Plattformen (Windows, Macintosh, Unix), auf welchen Sun Java Runtime 1.4 oder neuer läuft, den Snapform Viewer nutzen können. Das bedeutet für Windows: Windows ME, Windows 2000, Windows XP, Windows Vista. Für Macintosh OS X 10.1 und neuer.

Für Snapform Designer sind Einschränkungen vorhanden, welche eine Unterstützung unter Linux nicht gewähren. Für Snapform Designer ist entweder Windows (Varianten siehe oben) oder Macintosh OS X 10.2.8 oder neuer Voraussetzung.

Der geringe "Footprint" von Snapform Viewer und Designer verlangt daher auch keine besonderen Voraussetzungen bezüglich Arbeitsspeicher und Festplatten. Bei geringem Arbeitsspeicher ist allerdings die Arbeitsgeschwindigkeit eingeschränkt.

Für Snapform PDF Converter ist Windows 2000, Windows XP oder Windows Vista notwendig. Intel Macintosh mit virtuellen Maschinen funktionieren ebenfalls.

weitere Hardware

An weiterer Hardware ist, sofern papier-basierte Arbeitsabläufe geplant sind, ein Drucker mit einer Auflösung von mindestens 300 dpi notwendig (wenn Barcodes mit geringen Modul-Grössen eingesetzt werden, empfehlen sich 600 dpi.

Bei PKI-basierten Anwendungen ist gegebenenfalls ein Signaturkartenleser notwendig.

Auf Empfangsseite für papier-basierte Arbeitsabläufe sind Erfassungsgeräte für allfällig eingesetzte Barcode-Systeme notwendig. Die benötigte Leistung dieser Geräte hängt vom zu verarbeitenden Formularvolumen ab.

Systemsoftware

Wie oben erwähnt, benötigt Snapform Viewer und Designer Java Runtime 1.4.

Weitere Systemsoftware ist nicht notwendig.

Für die elektronische Datenübermittlung vom Formularen und Eingabedaten an ein zentrales Datensystem wird eine Internetverbindung vorausgesetzt. Dies ist jedoch anwendungsspezifisch.

2.2 Vorbereitung des Systems

Voraussetzung für die Installation von Snapform Designer und Snapform Viewer ist Sun Java Runtime Engine 1.4 oder höher. Diese Komponenten sind entweder bereits im Lieferumfang des Betriebssystems vorhanden, oder sie können von der Website von Sun (http://www.java.com) bzw. von Apple (http://www.apple.com) heruntergeladen werden.

2.3 Standardinstallation

Die Installation von Snapform Designer und Snapform Viewer verläuft dank geführter und einfach verständlicher Installationsroutine rasch und problemlos. Wenn Java Runtime auf dem Windows System nicht verfügbar ist, wird das Programm während der Installation automatisch heruntergeladen und vorkonfiguriert.

Wenn sowohl Snapform Designer als auch Snapform Viewer installiert werden sollen, ist es sinnvoll, erst den Designer und anschliessend den

Viewer zu installieren. Dies hat allerdings einzig einen Einfluss auf die Dateityp-Zuordnung.

2.3.1 Installation von Snapform Designer

Der aktuelle Installer von Snapform Designer kann jederzeit von der Snapform Website (http://www.snapform.com) heruntergeladen werden.

Es ist zu beachten, dass Snapform Designer mit Benutzer-Privilegien installiert werden kann, und keine Administratorenrechte benötigt.

Windows

Für Windows wird der Installer als .exe-Datei bereit gestellt. Nach dem Herunterladen kann der Installations-Wizard direkt gestartet werden. Die Installationsschritte sind im Wizard festgelegt und selbsterklärend.

Nach Durchlaufen der Installationsroutine ist Snapform Designer bereit.

Macintosh

Für Macintosh wird ein Disk Image im .dmg-Format bereit gestellt. Nach dem Mounten des Disk Images kann der Installer mit Doppelklick gestartet werden. Die Installationsschritte sind weitgehend selbsterklärend.

Nach Durchlaufen der Installationsroutine ist Snapform Designer bereit.

2.3.2 Installation von Snapform Viewer

Der aktuelle Installer von Snapform Viewer kann von der Snapform Website (**http://www.snapform.com**) heruntergeladen werden.

Es ist zu beachten, dass Snapform Viewer mit Benutzer-Privilegien installiert werden kann und keine Administratorenrechte benötigt.

Windows

Für Windows wird der Installer als .exe-Datei bereit gestellt. Nach dem Herunterladen kann der Installations-Wizard direkt gestartet werden. Die Installationsschritte sind im Wizard festgelegt und selbsterklärend.

Nach Durchlaufen der Installationsroutine ist Snapform Viewer bereit.

Macintosh

Für Macintosh wird ein Disk Image im .dmg-Format bereit gestellt. Nach dem Mounten des Disk Images kann der Installer mit Doppelklick gestartet werden. Die Installationsschritte sind weitgehend selbsterklärend.

Nach Durchlaufen der Installationsroutine ist Snapform Viewer bereit.

Linux

Für Linux wird ein Shell-Installer bereit gestellt. Nach dem Herunterladen wird der Installer ausgeführt. Die weiteren Installationsschritte sind weitgehend selbsterklärend.

Nach Durchlaufen der Installationsroutine ist Snapform Viewer bereit.

2.3.3 Installation von Snapform PDF Converter

Der Snapform PDF Converter wird als Installer im .exe-Format bereit gestellt. Nach dem Herunterladen kann der Installations-Wizard direkt gestartet werden. Die Installationsschritte sind im Wizard festgelegt und selbsterklärend.

Nach Durchlaufen der Installationsroutine ist Snapform PDF Converter bereit.

Tip: Beim PDF Converter handelt es sich um eine .NET-Anwendung. Wenn Sie eine neuere Version des PDF-Converters installieren möchten, sollten Sie daher vorgängig die bestehende Installation mittels Deinstallationsroutine entfernen.

2.4 Registrierung

Bei Snapform Viewer ist keine Registrierung notwendig. Dieser Abschnitt ist nur für Snapform Designer relevant.

Die heruntergeladene Version von Snapform Designer läuft ohne Registrierung im Demo-Modus.

Damit die lizenzierte Funktionalität freigeschaltet wrden kann, muss der Lizenzschlüssel eingegeben werden. Der Lizenzschlüssel erhalten Sie in der Regel von Ringler Informatik zugeschickt . Im Schlüssel sind auch die Informationen über den lizenzierten Funktionsumfang sowie die Anzahl Benutzer enthalten.

Mit Wahl von Menupunkt **Lizenzschlüssel eingeben** im Menu **Hilfe** öffnet sich das Lizenzschlüssel-Eingabefenster (siehe Fig. 2-1).

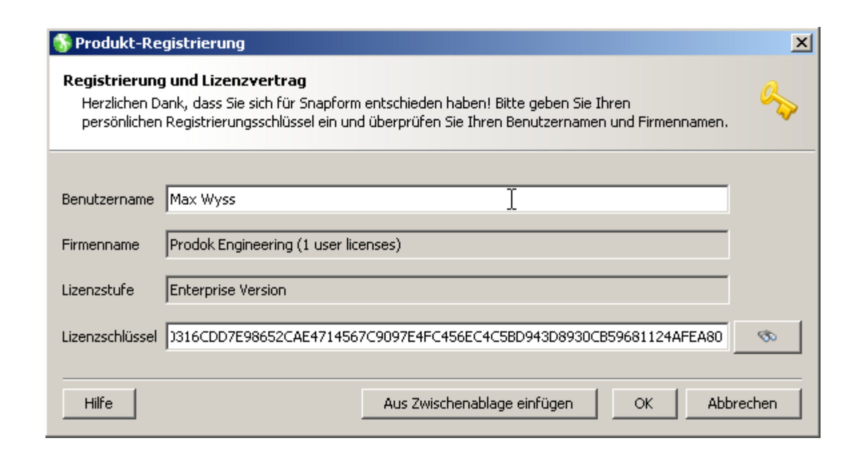


Fig. 2-1 *Eingabefenster für Lizenzschlüssel*

In diesem Fenster sind die Lizenzangaben einzugeben. Dabei beachten Sie bitte, dass **Benutzername** und **Firmenname** mit den Angaben bei der Bestellung des Lizenzschlüssels übereinstimmen.

Das Feld **Lizenzstufe** wird bei der Validierung des Lizenzschlüssels ausgefüllt.

Der Lizenzschlüssel wird am einfachsten über die Lizenzdatei geladen. Das Browse-Symbol rechts des Eingabefelds für den Lizenzschlüssel öffnet einen “Datei öffnen”-Dialog, in welchem die Lizenzdatei angegeben werden kann. Nach dem Bestätigen der Eingabe wird die Datei geöffnet und der Schlüssel automatisch registriert.

Es kann auch sein, dass der Lizenzschlüssel anderweitig übermittelt worden ist (z.B. als Textfolge innerhalb eines Emails). In diesem Fall empfiehlt es sich, den Schlüssel in die Zwischenablage zu kopieren und

mit Anklicken der Schaltfläche **Aus Zwischenablage einfügen** einzufügen.

Mit **OK** werden die Eingaben bestätigt und sofern die Lizenzinformationen korrekt sind, wird der Snapform Designer entsprechend der Lizenzstufe freigeschaltet.

2.5 Konfiguration

Snapform Designer braucht nicht weiter konfiguriert zu werden.

Die verschiedenen Sprachversionen heissen **Snapform Designer DE** (deutsch), **Snapform Designer EN** (englisch) sowie **Snapform Designer FR** (französisch) und können gezielt wie folgt angesprochen werden:

Windows: **Start** Menu —> **alle Programme** —> **Snapform**, dann Wahl der Sprachversion.

Macintosh: Im Installationsverzeichnis befinden sich die Programms-Smbole für die Sprachversionen. Diese Symbole können in das Dock gezogen werden.

2.6 Deinstallation

Im Installationsverzeichnis von Snapform Designer und Snapform Viewer befindet sich ein Deinstallationsprogramm.

Zum Deinstallieren ist dieses Programm aufzurufen. Dadurch werden alle betroffenen Komponenten entfernt.

3 Verwendung des Designers

3.1 Programm starten und beenden

3.1.1 Programmstart

Snapform Designer wird wie ein normales Programm gestartet. Das bedeutet, entweder aus dem **Start** Menu (unter Windows) bzw. dem Dock (OS X), oder aber durch Doppelklick auf das Programm-Icon oder mit Doppelklick auf das Symbol einer dem Snapform Designer zugeordneten Datei.

Hinweis: *Wenn Snapform Viewer ebenfalls auf dem System installiert ist, kann es sein, dass beim Doppelklick auf eine Datei des Typs* **.qdf** *bzw* **.qdft** *je nach Einstellung der Dateizuordnungen der Snapform Viewer geöffnet wird. In diesem Fall kann Snapform Designer nicht mittels Doppelklick auf ein Dateisymbol gestartet werden.*

Während des Programmstarts erscheint das Startfenster (Fig. 3-1).

Fig. 3-1 *Startfenster*



Das Startfenster zeigt die Versionsnummer, die Registrierinformation sowie den Hersteller.

Wenn das Programm vollständig geladen ist, erscheint die Arbeitsfläche (siehe Abschnitt 3.2.1).

Wenn der Startvorgang beendet ist, erscheint das leere Programmfenster, welches im weiter unten beschrieben ist (Abschnitt 3.2.1).

3.1.2 Programm beenden

Das Programm wird normalerweise über den Menubfehl **Datei --> Beenden** (Windows) bzw **Snapform Designer --> Beenden** beendet. Sind nicht gespeicherte Dokumente geöffnet, erscheint ein Popup mit der Frage nach dem Speichern.

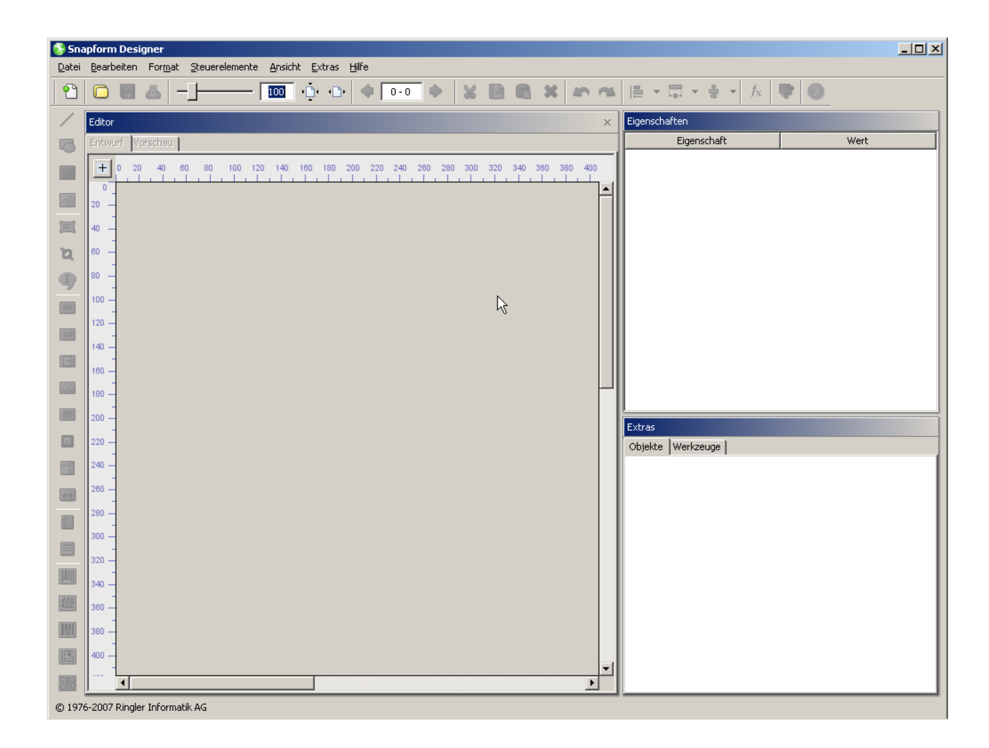
Das Programm kann ebenso mit Tastaturkürzeln und Schliessboxen beendet werden.

3.2 Komponenten von Snapform Designer

3.2.1 Das Programmfenster

Das leere Programmfenster ist in Fig. 3-2 dargestellt.

Fig. 3-2 *Programmfenster, leer*



Das Programmfenster umfasst die folgenden Bereiche:

1 Menuleiste und Menus
2 Werkzeugleiste
3 Controlsleiste
4 Arbeitsfläche
5 Eigenschaftenfenster
6 Werkzeugfenster

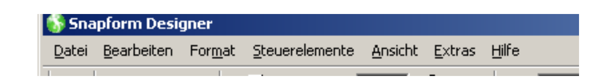
Diese Komponenten sind in den folgenden Abschnitten beschrieben.

3.2.2 Menuleiste

Die Menuleiste umfasst die wichtigsten Funktionen von Snapform Designer. Die Menupunkte sind kontextabhängig aktiv. Inaktive Elemente sind grau dargestellt und können auch nicht gewählt werden.

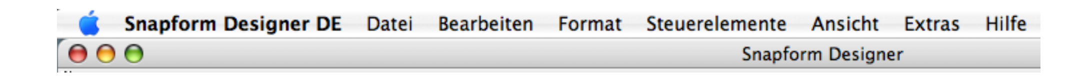
Die Menuleiste ist plattformabhängig. Die Windows-Version ist in Fig. 3-3 dargestellt, und die Macintosh-Version in Fig. 3-4.

Fig. 3-3 *Die Menuleiste der Windows-Version*



Die Menuleiste der Windows-Version ist im Wesentlichen eine Teilmenge der Menuleiste der Macintosh-Version. Es fehlen das Apfel-Menu sowie das Menu **Snapform Designer**.

Fig. 3-4 *Die Menuleiste der Macintosh-Version*



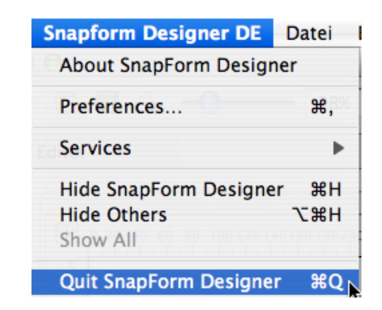
3.2.2.1 Das Apfel-Menu (Macintosh-spezifisch)

Unter dem Apfel-Menu befinden sich keine Snapform Designer-relevanten Menupunkte.

3.2.2.2 Das Menu Snapform Designer (Macintosh-spezifisch)

Das Menu **Snapform Designer** (siehe Fig. 3-5) ist spezifisch für die Macintosh-Implementation und fasst durch das System bereit gestellte Applikations-spezifische Menupunkte zusammen:

Fig. 3-5 *Das Menu Snapform Designer*



Hinweis: *Für die Screenshots wurde spezifisch die deutschsprachige Version von Snapform Designer benutzt. Daher heisst der Menupunkt* **Snapform Designer DE**.

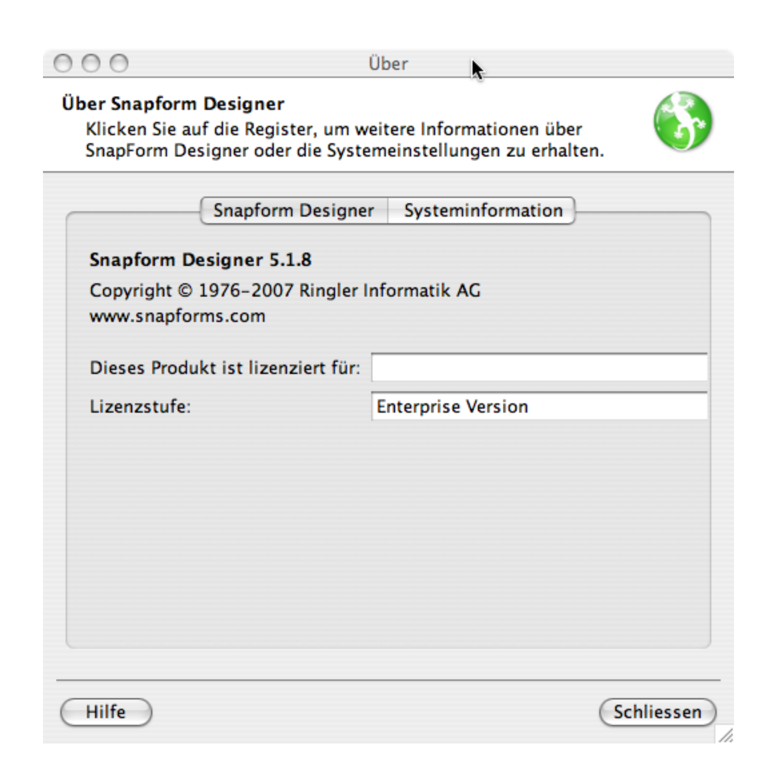
About Snapform Designer (nur Macintosh)

Mit Wahl dieses Menupunkts öffnet sich ein Informationsfenster zu Snapform Designer.

Der Menüpunkt **About Snapform Designer** zeigt die aktuelle Versionsinformation sowie die Lizenzangaben (siehe Fig. 3-6).

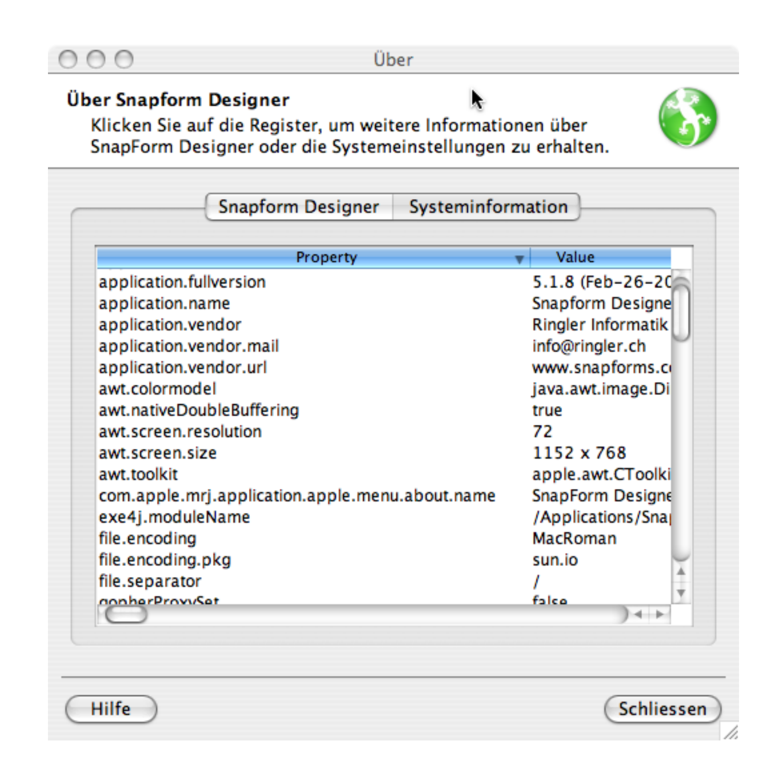
Hinweis: *Unter Windows ist dieser Menupunkt im Menu* **Hilfe** *angesiedelt.*

Fig. 3-6 *Das Über Snapform Designer Fenster*



Die Anzeige **Systeminformation** zeigt eine Übersicht über die für Snapform Designer relevante Systemumgebung an (siehe Fig. 3-7). Diese Information kann bei der Fehlersuche von grossem Nutzen sein.

Fig. 3-7 *Das Über System-information Fenster*



Preferences

Dies ist ein vom Betriebssystem bereit gestellter Platzhalter für ein Grundeinstellungs-Fenster. Snapform Designer besitzt allerdings keine anwenderseitig konfigurierbare Basiseinstellungen (gewisse Grundeinstellungen werden automatisch und kontextabhängig geführt).

Services

Dies ist ein Mac OS X-spezifischer Menupunkt, welcher die verschiedenen über das Betriebssystem bereitgestellten Dienste enthält. Die unter diesem Menupunkt vorhandenen Optionen sind rechnerabhängig und haben keinen direkten Bezug zu Snapform Designer.

Hide Snapform Designer

Blendet alle Fenster von Snapform Designer aus.

Hide others

Blendet die Fenster von allen anderen laufenden Applikationen ausser Snapform Designer aus.

Quit Snapform Designer

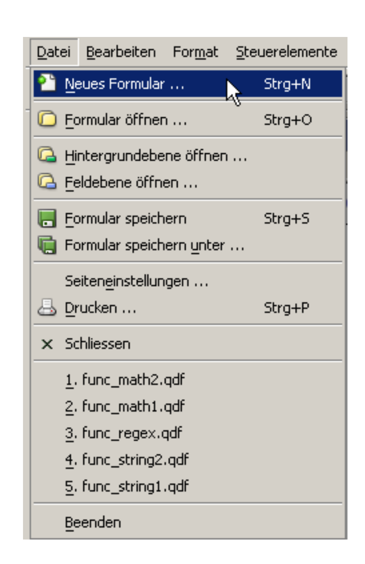
Beendet Snapform Designer.

Hinweis: *Unter Windows befindet sich der Befehl zum Beenden von Snapform Designer im Menu* **Datei**.

3.2.2.3 Das Menu Datei

Das Menu **Datei** (siehe Fig. 3-8) umfasst Datei-bezogene Befehle, wie das Öffnen und Speichern von Dokumenten.

Fig. 3-8 *Das Menu Datei*



Neues Formular

Öffnet das Spezifikationsfenster für ein neues Formular. Diese Funktion ist in Abschnitt 4.3.3 beschrieben

Formular öffnen

Öffnet einen "Datei Öffnen" Dialog, aus welchem das gewünschte **.qdf**-Dokument ausgewählt werden kann.

Wenn bereits ein Dokument offen ist, wird dieses erst geschlossen. Wenn ungespeicherte Änderungen vorhanden sind, erscheint ein Dialog, welcher fragt, ob das Dokument gespeichert werden soll (siehe Fig. 3-10).

Hintergrundebene öffnen

Öffnet einen "Datei Öffnen" Dialog, aus welchem das gewünschte Hintergrund-Dokument (im **.hdt** Format) ausgewählt werden kann.

Die Hintergrundebene wird in einem separaten Arbeitsschritt erzeugt, indem eine PDF-Datei mittels dem Snapform PDF-Converter in eine HDT-Datei umgewandelt wird.

Wenn bereits ein Dokument offen ist, wird dieses erst geschlossen. Wenn ungespeicherte Änderungen vorhanden sind, erscheint ein Dialog, welcher fragt, ob das Dokument gespeichert werden soll (siehe Fig. 3-10).

Feldebene öffnen

Öffnet einen "Datei Öffnen" Dialog, aus welchem das gewünschte Hintergrund-Dokument (im **.hdo** Format) ausgewählt werden kann.

Wenn bereits ein Dokument offen ist, wird dieses erst geschlossen. Wenn ungespeicherte Änderungen vorhanden sind, erscheint ein Dialog, welcher fragt, ob das Dokument gespeichert werden soll (siehe Fig. 3-10).

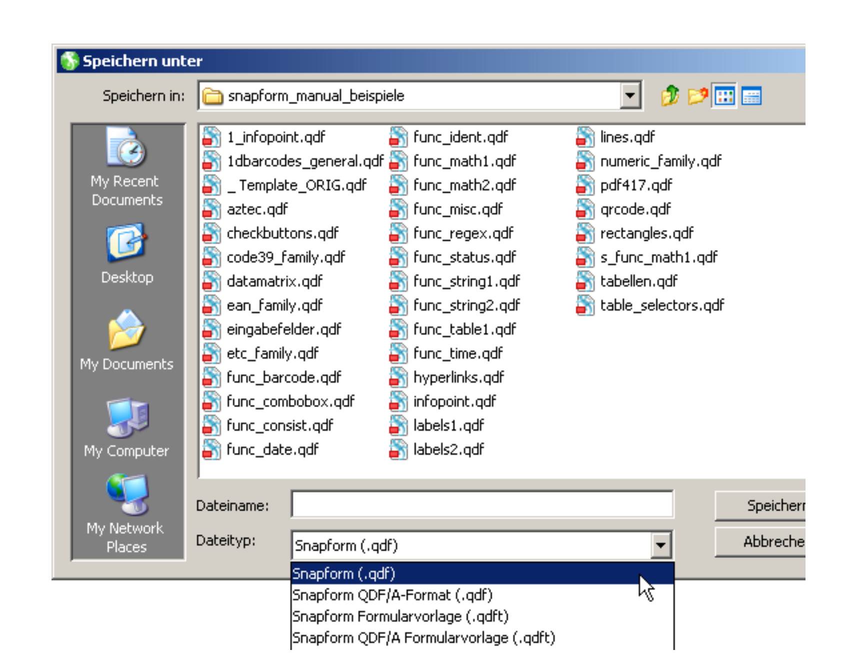
Formular speichern

Speichert das Formular unter dem bestehenden Namen. Dabei wird eine allfällige, bereits existierende Version überschrieben.

Formular speichern unter...

Öffnet einen "Speichern Unter…" Dialog, in welchem das Zielverzeichnis und der Dateiname eingegeben werden kann (siehe Fig. 3-9)

Fig. 3-9 *Der Speichern Unter… Dialog*



Es stehen die folgenden Dateiformate zur Auswahl:

- **Snapform (.qdf)**: das Standardformat von Snapform
- **Snapform QDF/A (.qdf)**: das Standardformat von Snapform mit offengelegter Datenebene
- **Snapform Formularvorlage (.qdft)**: Formular-Vorlage, welche sich immer ohne allfällig bereits gespeicherte Daten leer öffnet
- **Snapform QDF/A Formularvorlage (.qdft)**: Formular-Vorlage mit offengelegter Datenebene, welche sich immer ohne allfällig bereits gespeicherte Daten leer öffnet
- **Snapform Feldebene (.hdo)**: Format, welches nur die Feldebene enthält

Wenn keine Dateiendung angegeben ist, wird sie automatisch angefügt.

Seiteneinstellungen

Öffnet den normalen "Seiteneinstellungen" (oder "Seite einrichten…") Dialog, welcher vom Betriebssystem bereit gestellt wird.

Drucken

Öffnet den normalen vom Betriebssystem bereit gestellten Druckdialog.

Schliessen

Schliesst das Formular. Wenn ungesicherte Änderungen vorhanden sind, erscheint der Dialog zur Frage, ob das Dokument vorher gespeichert werden soll (siehe Fig. 3-10).



Fig. 3-10 *Der Dokument Schliessen Dialog*

1. bis **5.** zuletzt geöffnete Dateien

Diese Liste umfasst die 5 zuletzt geöffneten Dateien. Dabei kommen nur diejenigen Dateien in die Liste, welche mit dem **Datei Öffnen** Dialog geöffnet worden sind.

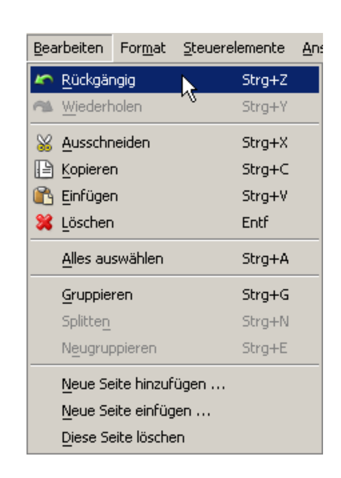
Beenden (nur unter Windows)

Beendet Snapform Designer. Dieser Befehl ist nur unter Windows in diesem Menu. Unter Mac OS X befindet sich dieser Befehl im Menu **Snapform Designer**.

3.2.2.4 Das Menu Bearbeiten

Im Menu **Bearbeiten** sind die Befehle zum Bearbeiten von Elementen (Kopieren, Ausrichten, etc.) zusammengefasst (siehe Fig. 3-11).

Fig. 3-11 *Das Menu Bearbeiten*



Rückgängig

Die letzte Aktion wird rückgängig gemacht.

Wiederholen

Die letzte Aktion wird wiederholt (auch “rückgängig machen von **Rückgängig**”)

Ausschneiden

Die selektierten Elemente werden in die Zwischenablage gelegt und von der Arbeitsfläche entfernt. Bei Formularelementen werden sämtliche Eigenschaften mitgegeben (ausser Tab-Order).

Kopieren

Die selektierten Elemente werden in die Zwischenablage gelegt. Bei Formularelementen werden sämtliche Eigenschaften mitkopiert (ausser Tab-Order).

Einfügen

Der Inhalt der Zwischenablage wird eingefügt (sofern dies möglich ist). Dabei werden sämtliche mitgegebenen Eigenschaften angewandt. Das

bedeutet, dass auch die Position der Elemente gleich wie beim Kopieren ist.

Löschen

Die selektierten Elemente werden ohne in die Zwischenablage gelegt zu werden entfernt.

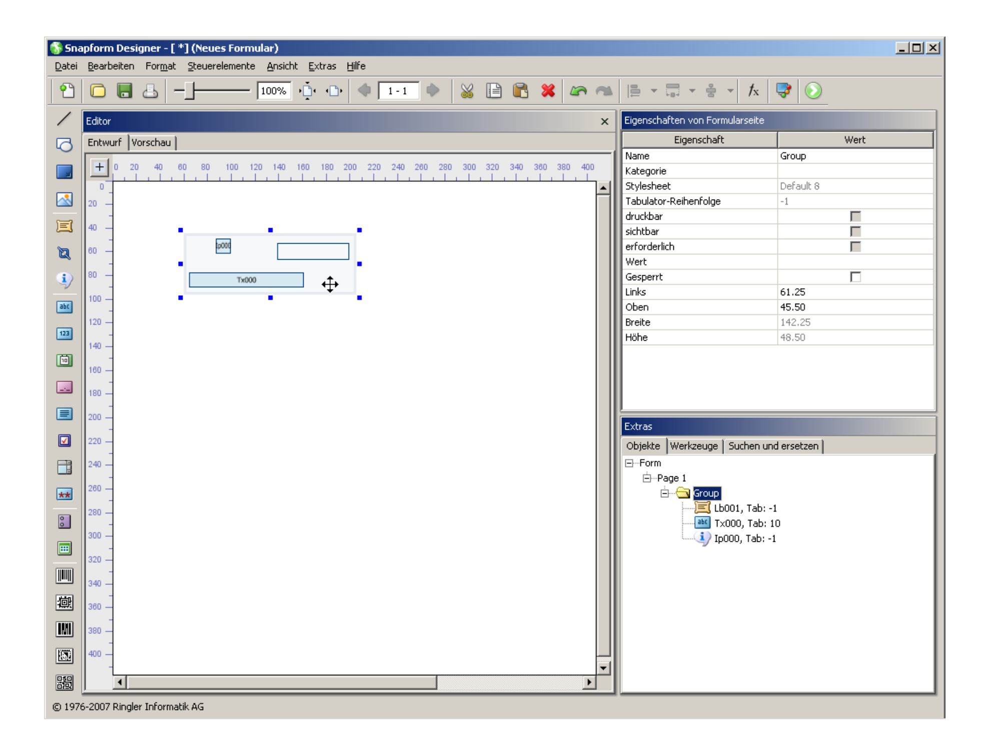
Alles auswählen

Sämtliche Elemente der Arbeitsfläche werden selektiert.

Gruppieren

Die selektierten Elemente werden zu einer Gruppe zusammengefasst. Die Gruppe erhält einen Default-Namen **Group** und kann zum Teil wie ein separates Formularelement behandelt werden (siehe Fig. 3-12).

Fig. 3-12 *Eine Gruppe*



Eine Gruppe ist ein Formularelement, in welchem andere Formularelemente zusammengefasst ('gruppiert') werden können. Der Hauptzweck der Gruppe ist, die relative Position der enthaltenen Formularelemente einzufrieren. Die folgenden Eigenschaften können für eine Gruppe verändert werden.

Name

Der Name der Gruppe im Dokument. Dieser Name kann geändert werden.

Gesperrt

Wenn das Kontrollkästchen gewählt ist, wird die Gruppe blockiert und kann nicht (versehentlich) verschoben werden. Zudem werden sämtliche übrigen Eigenschaften im Eigenschaften-Dialog deaktiviert. Der Zugriff auf die Eigenschaften ist erst wieder möglich, wenn dieses Kontrollkästchen deaktiviert wird.

Links

X-Koordinate der linken oberen Ecke der Umrandungsbox des Elements in Points. Diese Eigenschaft erlaubt eine genaue numerische Positionierung der Gruppe.

Oben

Y-Koordinate der linken oberen Ecke der Umrandungsbox des Elements in Points. Diese Eigenschaft erlaubt eine genaue numerische Positionierung der Gruppe.

Alle übrigen im Eigenschaften-Fenster angezeigten Eigenschaften sind nicht veränderbar.

In Gruppen enthaltene Elemente können über die Objektstruktur selektiert werden. Damit können deren Eigenschaften verändert werden. Allerdings sind die Positions-Eigenschaften (**Links**, **Oben**, **Breite**, **Höhe**) gesperrt. Diese Eigenschaften werden erst wieder zugänglich, wenn die Gruppe mit dem Befehl **Splitten** aufgelöst wird.

Splitten

Wenn das selektierte Element eine Gruppe ist, wird sie mit diesem Befehl wieder in Einzelemente aufgelöst.

Neugruppieren

Mit diesem Befehl kann die in der Selektion enthaltene Gruppe um neue Elemente innerhalb der Selektion erweitert werden.

Neue Seite hinzufügen

Fügt eine neue Seite am Ende des Formular hinzu. Diese Funktion ist in Abschnitt 4.3.4.1 beschrieben.

Neue Seite einfügen

Fügt eine neue Seite vor der aktuellen Seite hinzu. Diese Funktion ist in Abschnitt 4.3.4.1 beschrieben.

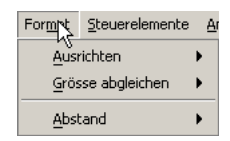
Diese Seite löschen

Entfernt die aktuelle Seite. Diese Funktion ist in Abschnitt 4.3.4.1 beschrieben.

3.2.2.5 Das Menu Format

Das Menu Format enthält Funktionen zum gegenseitigen Ausrichten von selektierten Formularelementen (siehe Fig. 3-13).

Fig. 3-13 *Das Menu Format*

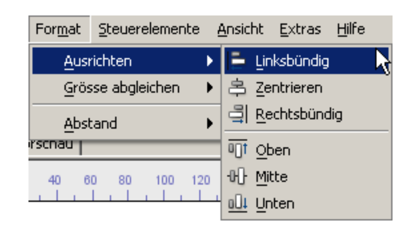


Das Menu ist nur dann aktiv, wenn mindestens zwei Formularelemente selektiert sind. Es besitzt drei Untermenus. Das aktive selektierte Element (mit den vollen schwarzen Markierungen der Umrandungsbox) gilt dabei als massgebende Referenz.

Untermenu **Ausrichten**

In diesem Untermenu (siehe Fig. 3-14) werden die selektierten Elemente gegenseitig ausgerichtet. Für die Ausrichtung massgebend ist die jeweilige Umrandungsbox.

Fig. 3-14 *Das Menu Format —> Ausrichten*



Die oberen drei Positionen steuern die horizontale Ausrichtung und die unteren drei Positionen die vertikale Ausrichtung.

Linksbündig

Die Elemente werden auf ihren linken Rand horizontal ausgerichtet.

Zentrieren

Die Elemente werden auf ihre Mitte horizontal ausgerichtet.

Rechtsbündig

Die Elemente werden auf ihren rechten Rand horizontal ausgerichtet.

Oben

Die Elemente werden auf ihren oberen Rand vertikal ausgerichtet.

Mitte

Die Elemente werden auf ihre Mitte vertikal ausgerichtet.

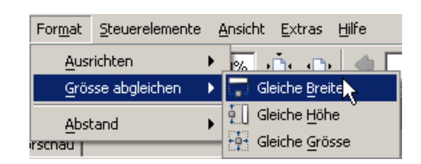
Unten

Die Elemente werden auf ihren unteren Rand vertikal ausgerichtet.

Untermenu **Grösse abgleichen**

In diesem Untermenu (siehe Fig. 3-15) werden Dimensionen der selektierten Elemente abgeglichen. Für den Abgleich massgebend ist die Umrandungsbox der Elemente.

Fig. 3-15 *Das Menu Format —> Grösse abgleichen*



Gleiche Breite

Alle selektierten Elemente erhalten die gleiche Breite wie das Referenzelement.

Gleiche Höhe

Alle selektierten Elemente erhalten die gleiche Höhe wie das Referenzelement.

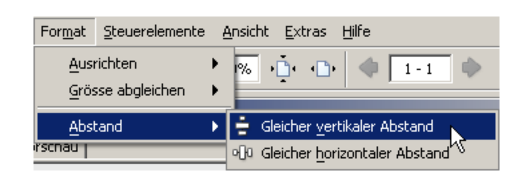
Gleiche Grösse

Alle selektierten Elemente erhalten die gleiche Breite und Höhe wie das Referenzelement.

Untermenu **Abstand**

In diesem Untermenu (siehe Fig. 3-16) werden die Abstände zwischen den selektierten Elementen abgeglichen. Massgebend ist dabei die jeweilige vertikale oder horizontale Mittelachse der Umrandungsbox der Elemente. Damit dieses Menu aktiv ist, müssen mindestens drei Elemente selektiert worden sein.

Fig. 3-16 *Das Menu Format —> Abstand*



Gleicher vertikaler Abstand

Die Elemente werden in vertikaler Richtung mit gleichem Abstand positioniert.

Gleicher horizontaler Abstand

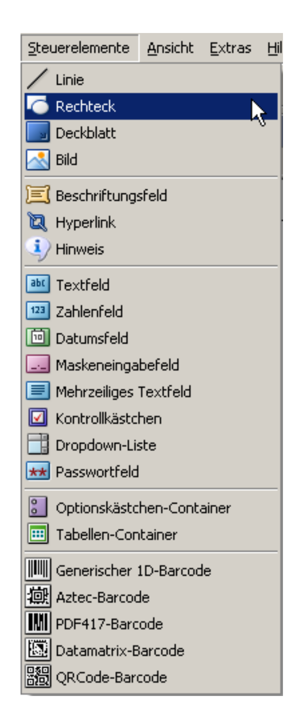
Die Elemente werden in horizontaler Richtung mit gleichem Abstand positioniert.

3.2.2.6 Das Menu Steuerelemente

Hinweis: *Der Begriff "Steuerelemente" ist eine Übersetzung des englischen Begriffs "Controls", der in den verschiedenen Normen im Zusammenghang mit Formularementen benutzt wird. In diesem Handbuch wird für "Steuerelement" auch häufig der Begriff "Werkzeug" benutzt.*

Das Menu **Steuerelemente** (siehe Fig. 3-17) enthält alle Werkzeuge zum Erzeugen von Formularelementen.

Fig. 3-17 *Das Menu Steuerelemente*



Die Liste der Menuelemente entspricht der vertikalen Werkzeugleiste (siehe 3.2.4).

Linie	Das Linien-Werkzeug ist in Abschnitt 4.3.5.1 beschrieben.
Rechteck	Das Rechteck-Werkzeug ist in Abschnitt 4.3.5.2 beschrieben.
Deckblatt	Das Abdeckflächen-Werkzeug ist in Abschnitt 4.3.5.3 beschrieben.
Bild	Das Bild-Werkzeug ist in Abschnitt 4.3.5.4 beschrieben.
Beschriftungsfeld	Das Beschriftungsfeld-Werkzeug ist in Abschnitt 4.3.6.1 beschrieben.
Hyperlink	Das Hyperlink-Werkzeug ist in Abschnitt 4.4.8.1 beschrieben.
Hinweis	Das Hinweis-/Infopunkt-Werkzeug ist in Abschnitt 4.4.8.2 beschrieben.
Textfeld	Das Textfeld-Werkzeug ist in Abschnitt 4.4.2.2 beschrieben.
Zahlenfeld	Das Zahlenfeld-Werkzeug ist in Abschnitt 4.4.2.4 beschrieben.
Datumsfeld	Das Datumsfeld-Werkzeug ist in Abschnitt 4.4.2.5 beschrieben.
Maskeneingabefeld	Das Maskeneingabefeld-Werkzeug ist in Abschnitt 4.4.2.6 beschrieben.
Mehrzeiliges Textfeld	Das Mehrzeilenfeld-Werkzeug ist in Abschnitt 4.4.2.3 beschrieben.
Kontrollkästchen	Das Kontrollkästchen-Werkzeug ist in Abschnitt 4.4.3.1 beschrieben.
Dropdown-Liste	Das Listenfeld-Werkzeug ist in Abschnitt 4.4.4 beschrieben.
Passwortfeld	Das Passwortfeld-Werkzeug ist in Abschnitt 4.4.2.7 beschrieben.

Optionskästchen-Container

Das Optionsfelder-Werkzeug ist in Abschnitt 4.4.3.2 beschrieben.

Tabellen-Container

Das Tabellen-Werkzeug ist in Abschnitt 4.4.7.1 beschrieben.

Generischer 1D-Barcode

Das 1D-Barcode-Werkzeug ist in Abschnitt 4.4.5.1 beschrieben.

Aztec-Barcode

Das Aztec-Barcode-Werkzeug ist in Abschnitt 4.4.6.2 beschrieben.

PDF417-Barcode

Das PDF417-Barcode-Werkzeug ist in Abschnitt 4.4.6.3 beschrieben.

Datamatrix-Barcode

Das Datamatrix-Barcode-Werkzeug ist in Abschnitt 4.4.6.4 beschrieben.

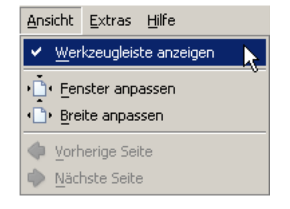
QRCode-Barcode

Das QRCode-Barcode-Werkzeug ist in Abschnitt 4.4.6.5 beschrieben.

3.2.2.7 Das Menu Ansicht

Das Menu Ansicht (siehe Fig. 3-18) umfasst Funktionen zur Ansicht des Formulars auf der Arbeitsfläche und zum Seitenwechsel.

Fig. 3-18 *Das Menu Ansicht*



Werkzeugleiste anzeigen

Mit diesem Umschalter wird die Werkzeugleiste (siehe Abschnitt 3.2.3) ein- und ausgeblendet.

Fenster anpassen

Skaliert das aktive Dokument derart, dass es vollständig im Editor-Fenster angezeigt wird.

Breite anpassen

Skaliert das aktive Dokument derart, dass es in seiner Breite im Editor-Fenster angezeigt wird.

Vorherige Seite

Wechselt zur vorangehenden Seite (nur aktiv, wenn das Dokument mehrere Seiten umfasst).

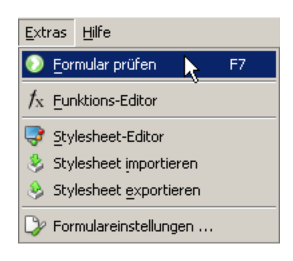
Nächste Seite

Wechselt zur nächsten Seite (nur aktiv, wenn das Dokument mehrere Seiten umfasst).

3.2.2.8 Das Menu Extras

Im Menu **Extras** (siehe Fig. 3-19) sind Spezialfunktionen zusammengefasst, welche bei der Arbeit mit Formularen wichtig sein können.

Fig. 3-19 *Das Menu Extras*



Formular prüfen

Prüft das Formular auf Inkonsistenzen und nicht zulässige Formel-Syntax und teilt dies mit (siehe Abschnitt 4.6).

Funktions-Editor

Öffnet den Funktionseditor. Der Funktionseditor ist im Abschnitt 4.5.1.2 beschrieben.

Stylesheet-Editor

Öffnet den Stylesheet-Editor. Der Stylesheet-Editor ist im Abschnitt 4.7.1 beschrieben.

Stylesheet importieren

Öffnet einen Datei Öffnen-Dialog zur Auswahl einer Formulardatei (**.qdf**) oder einer Stylesheet-Datei (**.qdl**), aus welcher die Stylesheets importiert werden (siehe Abschnitt 4.7.2).

Stylesheet exportieren

Öffnet einen Datei Speichern Dialog und exportiert die Stylesheets des aktiven Dokuments in eine Stylesheet-Datei (**.qdl**) (siehe Abschnitt 4.7.2).

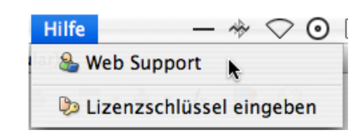
Formulareinstellungen

Öffnet das **Formulareinstellungen** Fenster (siehe Abschnitt 4.8) zur Einstellung von verschiedenen Formular- und Sicherheits-Optionen des aktiven Formulars.

3.2.2.9 Das Menu Hilfe

Das Menu Hilfe (siehe Fig. 3-20) enthält die Funktionen zur Benutzung von Snapform Designer.

Fig. 3-20 *Das Menu Hilfe*



Web Support

Dies ist der Link zur Supportseite auf der Snapform-Website: (**http://support.snapform.com/help/designer.asp?language=DE**)

Lizenzschlüssel eingeben

Öffnet das Fenster zur Eingabe des Lizenzschlüssels (siehe Abschnitt 2.4)

Über Snapform Designer (nur Windows)

Mit Wahl dieses Menupunkts öffnet sich ein Informationsfenster zu Snapform Designer. Dieses Fenster hat zwei Anzeigen. Die Anzeige **Snapform Designer** zeigt die Versionsinformation sowie die Lizenzangaben (siehe Fig. 3-6).

Die Anzeige **Systeminformation** zeigt eine Übersicht über die für Snapform Designer relevante Systemumgebung an (siehe Fig. 3-7). Diese Information kann bei der Fehlersuche von grossem Nutzen sein.

3.2.3 Werkzeugleiste

Die Werkzeugleiste (siehe) umfasst Werkzeug-Symbole für die wichtigen Aktionen mit den Formularen. Zum grössten Teil sind die Symbole auch als Menu-Aktion vorhanden.

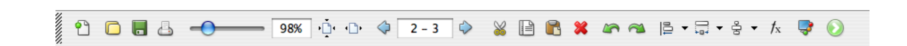


Fig. 3-21 *Die Werkzeugleiste*

Die einzelnen Werkzeuge sind nachstehend erklärt

Abreissleiste

Die Werkzeugleiste kann an der Abreissleiste “abgerissen” werden und als eigenständiges Fenster auf dem Bildschirm verschoben werden. Wenn dieses Fenster geschlossen wird, erscheint die Werkzeugleiste wieder im Hauptfenster.

Neues Formular

Mit diesem Symbol öffnet sich das Spezifikationsfenster für ein neues Formular. Diese Funktion ist in Abschnitt 4.3.3 beschrieben

Formular öffnen

Mit diesem Symbol öffnet sich ein Datei Öffnen Dialog, aus welchem das gewünschte **.qdf**-Dokument ausgewählt werden kann.

Wenn bereits ein Dokument offen ist, wird dieses erst geschlossen. Wenn ungespeicherte Änderungen vorhanden sind, erscheint ein Dialog, welcher fragt, ob das Dokument gespeichert werden soll (siehe Fig. 3-10).

Formular speichern



Mit diesem Symbol wird das Formular unter dem bestehenden Namen gespeichert. Dabei wird eine alfällige bereits existierende Version überschrieben.

Formular drucken



Mit diesem Symbol wird der normale vom Betriebssystem bereit gestellte Druckdialog geöffnet.

Zoomregler



Mit diesem kombinierten Symbol wird der Zoomfaktor für die Darstellung des Formulars im Editorfenster eingestellt. Mit dem Verschieben des Reglerknopfs ändert sich die numerische Anzeige des Zoomfaktors und die Darstellung des Formulars passt sich entsprechend an.

Es ist ebenfalls möglich, einen Prozentwert direkt einzugeben. Der Zoombereich läuft von **25%** bis **400%**.

Fenster anpassen

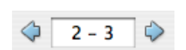


Dieses Symbol skaliert das aktive Dokument derart, dass es vollständig im Editor-Fenster angezeigt wird.

Breite anpassen 

Dieses Symbol skaliert das aktive Dokument derart, dass es in seiner Breite im Editor-Fenster angezeigt wird.

Seitennavigation



Mit diesem kombinierten Symbol wird durch ein mehrseitiges Dokument navigiert. Anklicken der Pfeilsymbole wird auf die vorangehende bzw. nächste Seite gewechselt. Die aktuelle Seite und die totale Anzahl Seiten sind im Anzeigefeld zwischen den Pfeilsymbolen angegeben.

Es ist ebenfalls möglich die Seitennummer direkt in das Anzeigefeld einzugeben.

Ausschneiden 

Die selektierten Elemente werden in die Zwischenablage gelegt und von der Arbeitsfläche entfernt. Bei Formularelementen werden sämtliche Eigenschaften mitgegeben (ausser Tab-Order).

Kopieren 

Die selektierten Elemente werden in die Zwischenablage gelegt. Bei Formularelementen werden sämtliche Eigenschaften mitgegeben (ausser Tab-Order).

Einfügen 

Der Inhalt der Zwischenablage wird eingefügt (sofern dies möglich ist). Dabei werden sämtliche mitgegebenen Eigenschaften angewandt. Das bedeutet, dass die Position der Elemente gleich wie beim Kopieren ist.

Löschen — Die selektierten Elemente werden, ohne in die Zwischenablage gelegt zu werden, entfernt.

Rückgängig  — Die letzte Aktion wird rückgängig gemacht.

Wiederholen  — Die letzte Aktion wird wiederholt (auch "rückgängig machen von **Rückgängig**")

Die nächsten drei Symbole sind Aufklappsymbole, welche wie Untermenus funktionieren. Die zuletzt gewählte Option bleibt im Symbol bestehen und kann durch Anklicken direkt wieder ausgeführt werden.

Aufklappsymbol **Ausrichten**



Mit diesem Aufklappsymbol öffnet sich dasselbe Untermenu wie beim Menu **Format —> Ausrichten** (siehe Fig. 3-14). Hier werden die selektierten Elemente gegenseitig ausgerichtet. Für die Ausrichtung massgebend ist die jeweilige Umrandungsbox.

Die oberen drei Positionen steuern die horizontale Ausrichtung und die unteren drei Positionen die vertikale Ausrichtung.

Linksbündig — Die Elemente werden auf ihren linken Rand horizontal ausgerichtet.

Zentrieren — Die Elemente werden auf ihre Mitte horizontal ausgerichtet.

Rechtsbündig — Die Elemente werden auf ihren rechten Rand horizontal ausgerichtet.

Oben — Die Elemente werden auf ihren oberen Rand vertikal ausgerichtet.

Mitte — Die Elemente werden auf ihre Mitte vertikal ausgerichtet.

Unten — Die Elemente werden auf ihren unteren Rand vertikal ausgerichtet.

Aufklappsymbol

Grösse abgleichen

Mit diesem Aufklappsymbol öffnet sich dasselbe Untermenu wie beim Menu **Format —> Grösse abgleichen** (siehe Fig. 3-15). Hier werden Dimensionen der selektierten Elemente abgeglichen. Für den Abgleich massgebend ist die Umrandungsbox der Elemente.

Gleiche Breite

Alle selektierten Elemente erhalten die gleiche Breite wie das Referenzelement.

Gleiche Höhe

Alle selektierten Elemente erhalten die gleiche Höhe wie das Referenzelement.

Gleiche Grösse

Alle selektierten Elemente erhalten die gleiche Breite und Höhe wie das Referenzelement.

Aufklappsymbol

Abstand

Mit diesem Aufklappsymbol öffnet sich dasselbe Untermenu wie beim Menu **Format —> Abstand** (siehe Fig. 3-16). Hier werden die Abstände zwischen den selektierten Elementen abgeglichen. Massgebend ist dabei die jeweilige vertikale oder horizontale Mittelachse der Umrandungsbox der Elemente. Damit dieses Menu aktiv ist, müssen mindestens drei Elemente selektiert worden sein.

Gleicher vertikaler Abstand

Die Elemente werden in vertikaler Richtung mit gleichem Abstand positioniert.

Gleicher horizontaler Abstand

Die Elemente werden in horizontaler Richtung mit gleichem Abstand positioniert.

Funktionseditor

Öffnet den Funktionseditor. Der Funktionseditor ist in Abschnitt 3.2.8 und Abschnitt 4.5.1.2 beschrieben.

Stylesheet-Editor



Öffnet den Stylesheet-Editor. Der Stylesheet-Editor ist im Abschnitt 4.7.1 beschrieben.

Formular prüfen 

Prüft das Formular auf Inkonsistenzen und nicht zulässige Formel-Syntax und teilt dies mit (siehe Abschnitt 4.6).

3.2.4 Steuerelemente-Leiste

Die Steuerelemente-Leiste entspricht in ihrem Aufbau exakt dem Menu **Steuerelemente**. Für die Elemente der Steuerelemente-Leiste wird daher auf Abschnitt 3.2.2.6 verwiesen.

3.2.5 Arbeitsfläche/Editor-Fenster

Die Arbeitsfläche (siehe Fig. 3-22), auch **Editor**-Fenster genannt, ist derjenige Bereich, in welchem Formulare gestaltet werden. Die Arbeitsfläche entspricht dem beim Erstellen einer Seite festgelegten Papierformat.

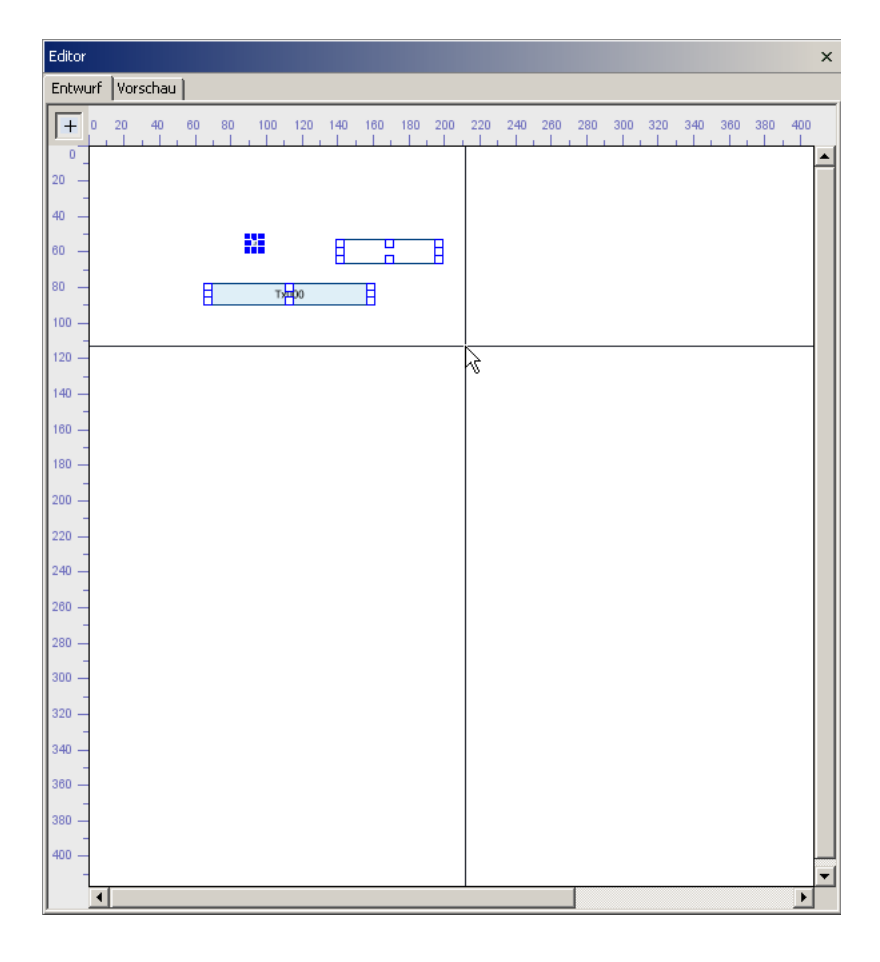


Fig. 3-22 *Das Editor-Fenster*

Das Editor-Fenster kann zwischen Entwurf-Ansicht und Vorschau umgeschaltet werden (siehe Abschnitt 3.3).

Ausgehend vom Nullpunkt in der oberen linken Ecke laufen Massstäbe mit Masseinheit **point**.

Mit Anklicken des Fadenkreuz-Symbols wird ein Fadenkreuz ein- und ausgeschaltet, welches die Positionierung von Elementen sowie das Vermessen von Positionen auf einem Hintergrundbild erleichtern kann. Im oben stehenden Bild ist das Fadenkreuz eingeschaltet.

Das Schliess-Symbol in der rechten oberen Ecke hat die gleiche Funktion wie der Menubefehl **Datei —> Schliessen**.

3.2.6 Eigenschaftenfenster

Je nach aktivem Werkzeug und aktiver Selektion zeigt das Eigenschaftenfenster (siehe Fig. 3-23. für ein repräsentatives Beispiel) unterschiedliche Informationen an.

Fig. 3-23 *Repräsentatives Beispiel für ein Eigenschaftenfenster*

Eigenschaften von Mehrzeiliges Textfeld

Eigenschaft	Wert
Name	MI000
Kategorie	
Stylesheet	Default 10
Tabulator-Reihenfolge	10
druckbar	☑
sichtbar	☑
erforderlich	☐
Wert	
Gesperrt	☐
Links	82.7922
Oben	45.7792
Breite	148.898
Höhe	61.00
schreibgeschützt	☐
Einmalfeld	☐
Kurzinfo-Text	
Maximale Zeilenzahl	0
Einzelne Zeile zentrieren	☐
Link zu Objekt	

Grundsätzlich zeigt das Eigenschaftenfenster Eigenschaftsname-/Eigenschaftswert-Paare an. Je nach Art der Eigenschaft wird der Wert über ein Text- oder Zahlenfeld eingegeben, über ein Kontrollkästchen

ein- und ausgeschaltet, oder aus einer Liste ausgewählt. Bei gewissen Eigenschaften (z.B. Wert) öffnet sich ein spezielles Editorfenster.

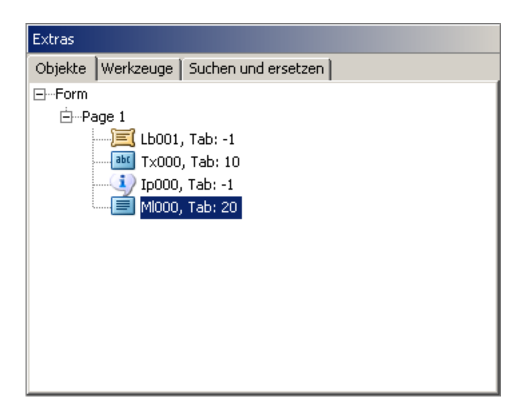
3.2.7 Extras-Fenster

Das **Extras**-Fenster dient zur Anzeige von zusätzlichen Informationen und Werkzeugen. Über einen Umschalter kann die Objektstruktur, ein Werkzeugfenster und ein Suchen-/Ersetzen-Fenster angezeigt werden.

3.2.7.1 Objekte

Wennn **Objekte** gewählt wird, zeigt das **Extras**-Fenster die Objektstruktur des Formulars (siehe Fig. 3-24). Dies ist auch die übliche Einstellung des Extras-Fenster.

Fig. 3-24 *Objektstruktur des Formulars*



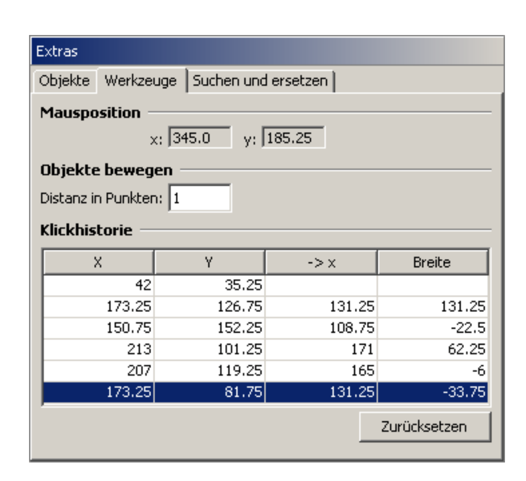
Die Objektstruktur ist hierarchisch gegliedert. Wenn Elemente ihrerseits Elemente enthalten, wird dies durch eine Aufklappmarke angezeigt. Mit der Aufklappmarke kann das Element auf- und zugeklappt werden, um entsprechend mehr Übersichtlichkeit zu schaffen.

Ein einzelnes Element oder eine Gruppe von Elementen kann in der Objektstruktur gewählt werden. Das Eigenschaftenfenster wird aktualisiert, und Eigenschaften können definiert werden.

3.2.7.2 Werkzeuge

Das **Werkzeuge**-Fenster enthält verschiedene Einstellungen, welche die Arbeit an den Formularen erleichtern (siehe Fig. 3-25).

Fig. 3-25 *Das Werkzeuge-Fenster*



In der Zone **Mausposition** wird fortlaufend die aktuelle Position des Mauscursors angezeigt.

In der Zone **Objekte bewegen** kann die Schrittgrösse (in **points**) eingegeben werden, um welche Objekte mit den Cursorsteuertasten bewegt werden. Wenn gleichzeitig die Umschalttaste gedrückt wird, bewegen sich die Objekte um die 10-fache Schrittlänge.

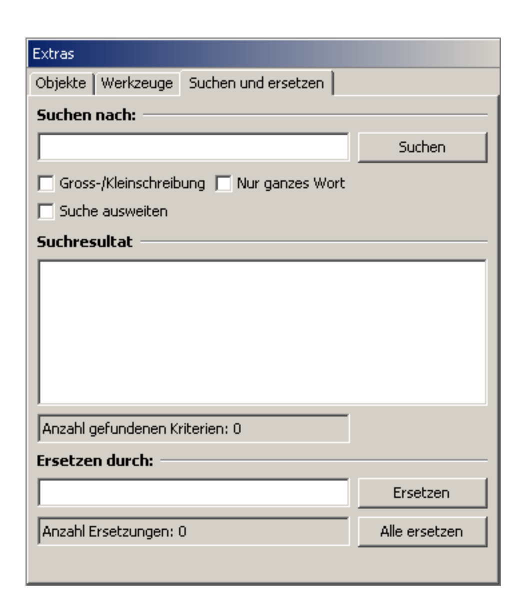
Wenn beim Klicken auf der Arbeitsfläche die Control-Taste gedrückt wird, wird die Mausposition in die Tabelle **Klickhistorie** eingetragen. Mit jedem fortlaufenden Eintrag wird zudem in der Spalte **–> x** der Abstand in X-Richtung vom ersten Punkt angegeben, und in der Spalte **Breite** der Abstand in X-Richtung vom vorhergehenden Punkt.

Mit Anklicken der Schaltfläche **Zurücksetzen** wird die **Klickhistorie** gelöscht.

3.2.7.3 Suchen und ersetzen

Mit der Wahl von **Suchen und Eresetzen** zeigt sich das **Suchen/ Ersetzen**-Fenster (siehe Fig. 3-26).

Fig. 3-26 *Das Suchen / Ersetzen-Fenster*



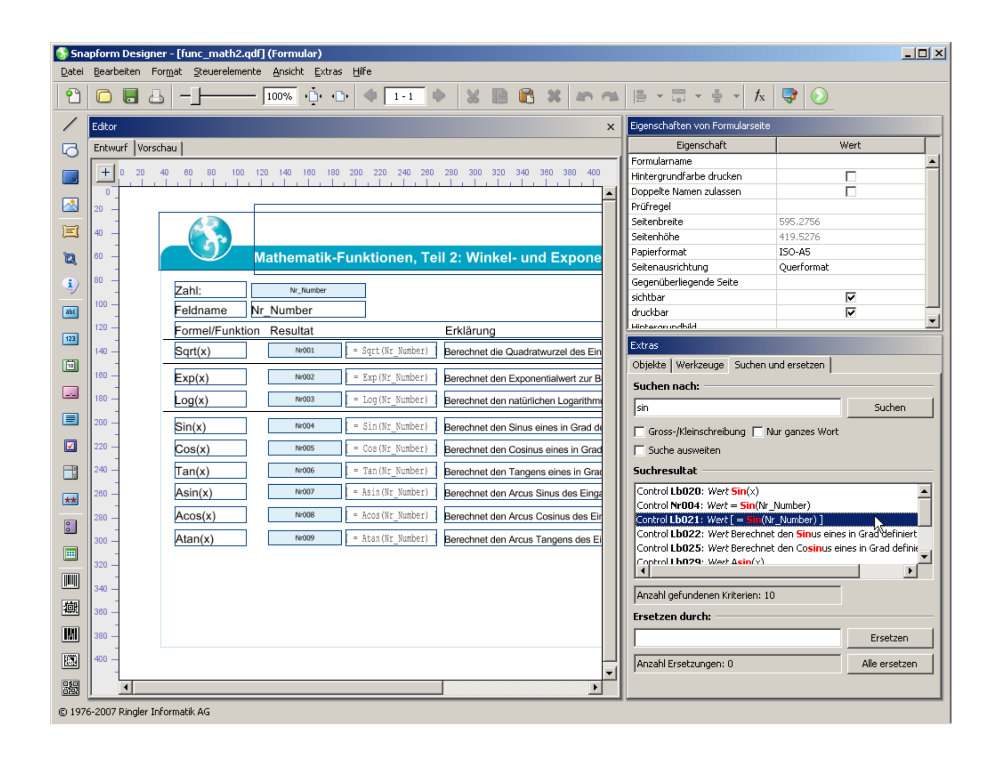
Im Eingabefeld **Suchen nach** wird eine Zeichenkette eingegeben. Mit Anklicken von **Suchen** wird der Suchvorgang ausgelöst.

In Normalfall werden Zeichenketten in den Eigenschaften **Name**, **Wert** und **Kurzinfo** gesucht. Wenn das Kontrollkästchen Suche ausweiten angeklickt ist, werden sämtliche Felder und deren Inhalte durchsucht.

Mit den Kontrollkästchen **Gross-/Kleinschreibung** und **Nur ganzes Wort** kann der Suchvorgang fein abgestimmt werden. Im ersten Fall wird die Gross-/Kleinschreibung berücksichtigt, und im zweiten Fall muss im zu durchsuchenden Text die Zeichenkette von "Worttrennern" begrenzt sein, damit eine Funstelle erkannt wird. Besonders wenn die Zeichenkette, nach der gesucht wird, nicht allzu aufschlussrieich ist, ist eine Verfeinerung nützlich.

Die Fundstellen werden im Feld **Suchresultat** zeilenweise angezeigt (siehe Fig. 3-27).

Fig. 3-27 *Suchresultate im Suchen / Ersetzen Fenster*



In diesem Beispiel wurde nach der Zeichenfolge **sin** gesucht. Sie kommt in verschiedenen Feldern vor. In der Zusammenstellung wird der Suchstring rot markiert. Mit Doppelklick auf eine Fundstelle wird das betreffende Element aktiv, und kann nun weiter bearbeitet werden.

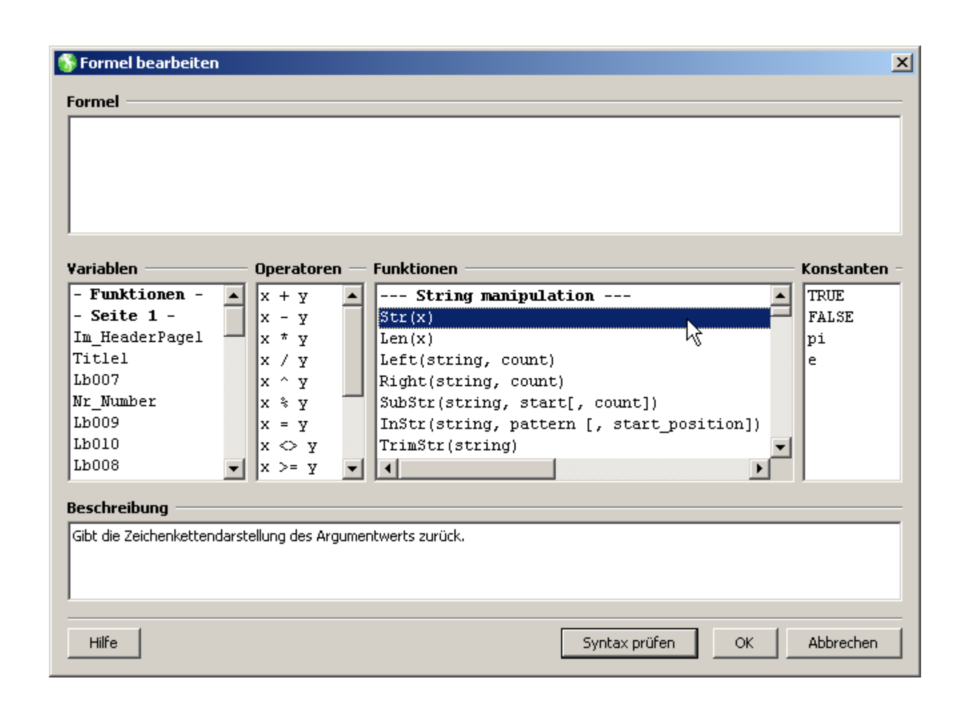
Im Feld **Ersetzen** durch wird die Zeichenkette eingegeben, durch welche die Fundstellen ersetzt werden sollen. Mit Anklicken von **Ersetzen** bzw. **Alle ersetzen** werden die Fundstellen ersetzt.

Hinweis: *Besonders wenn nach Zeichenketten gesucht wird, welche Formelfunktionen entsprechen oder Bestandteil von Formelfunktionen sind, ist bei Alle ersetzen aufzupassen, dass keine fehlerhaften Formeln entstehen.*

3.2.8 Formeleditor

Für den Aufbau von intelligenten elektronischen Formularen sind Formeln eine der wichtigsten Komponenten. Formeln werden im Formeleditor (siehe Fig. 3-28) zusammengestellt. Funktionen werden im identischen Funktionseditor zusammengestellt.

Fig. 3-28 *Der Formeleditor*



Das Eingabefeld **Formel** ist das eigentliche Arbeitsfeld, worin die Formel eingegeben wird.

Eine Formel beginnt immer mit dem Gleichheitszeichen (**=**). Wenn das Gleichheitszeichen fehlt, wird die Eingabe als wortwörtlicher Text betrachtet, und gegebenenfalls dargestellt.

Die Formel kann entweder direkt eingetippt, oder aber aus den Bausteinen von den Feldern **Variablen**, **Operatoren** und **Funktionen** zusammengestellt werden. Ein Doppelklick auf den betreffenden Baustein bringt ihn an die Eingabeposition im Feld **Formel**.

Die Bausteine haben Platzhalter für Argumente. Diese Platzhalter können mit Doppelklick selektiert werden, und ein Doppelklick auf den einzufügenden Baustein ersetzt den Platzhalter mit dem gewünschten Eintrag.

Im Feld **Beschreibung** erscheint für die eingebauten Bausteine ein kurzer Text, welcher zum grossen Teil mit den Beschreibungen im Abschnitt 5.3 übereinstimmt.

Auf diese Art können auch sehr komplexe Formeln mit einem Minimum an Tippaufwand zusammengestellt werden. Einzig bei Funktionen mit optionalen Argumenten ist darauf zu achten, dass die eckigen Klammern (**[,]**), welche diese Argumente kennzeichnen, entfernt werden.

Wenn die Formel fertig zusammengestellt ist, kann sie mit **OK** übernommen werden. Es wird allerdings empfohlen, zuerst die Formel auf formale Richtigkeit zu prüfen, mit Anklicken von **Syntax prüfen**. Wenn die Syntaxprüfung einen Fehler entdeckt, erscheint eine Fehlermeldung, welche Hinweise auf den Fehler gibt. Es ist jedoch immer möglich, die fehlerhafte Formel zu akzeptieren; sie muss einfach zu einem späteren Zeitpunkt korrigiert werden.

Getestet werden Formeln in der Vorschau (siehe Abschnitt 3.3.2). Um wirklich sicher zu sein, dass die Formeln korrekt funktionieren, sind Tests in Snapform Viewer notwendig.

3.2.9 Weitere Komponenten der Benutzeroberfläche

Der Snapform Designer hat weitere Benutzungselemente, welche an verschiedenen Orten eingesetzt werden können.

3.2.9.1 Tastaturkürzel

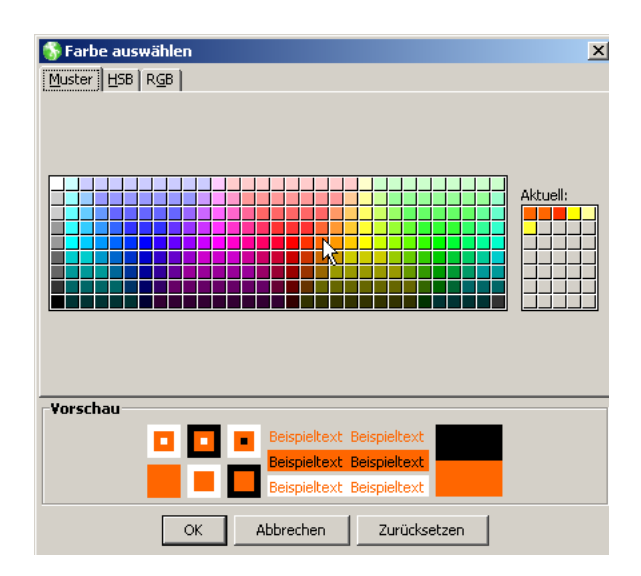
Besonders für den erfahrenen Benutzer erweisen sich Tastaturkürzel als effizientes Mittel, die Arbeit zu beschleunigen. Snapform Designer unterstützt Tastaturkürzel. Die betreffenden Tastenkombinationen sind in den jeweiligen Menus angegeben.

3.2.9.2 Farbauswahlfenster

Einige Feldeigenschaften bestehen aus einer Farbe. Zur Auswahl einer Farbe besitzt Snapform Designer einen Farbauswahl-Mechanismus, welcher über das Farbauswahlfenster (siehe Fig. 3-29) funktioniert. Zur Auswahl von Farben stehen drei verschiedene Farbmodelle zur

Verfügung. In der Ansicht **Muster** (siehe Fig. 3-29) erfolgt die Wahl aus einer Palette von sogenannten "web safe"-Farben. In der Ansicht **HSB** (siehe Fig. 3-30) wird das HSB-Farbmodell zur Auswahl und Definition von Farben benutzt, und in der Ansicht **RGB** (siehe Fig. 3-31) können die RGB-Werte der gewünschten Farbe festgelegt werden.

Fig. 3-29 *Farbauswahlfenster mit Farbmuster*

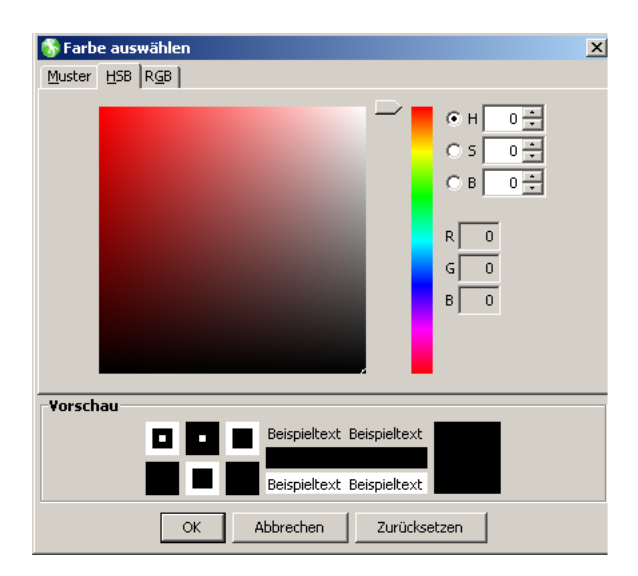


Das Farbauswahlfenster wird mit Anklicken auf den Farbbalken im Eigenschaften-Fenster geöffnet. In der Auswahl mit **Farbmuster** wird die gewünschte Farbe mit Anklicken gewählt. Für jedes der Felder im Muster gibt der Kurztext des Mauscursors den RGB-Wert der zugehörigen Farbe an.

Im Bereich **Vorschau** wird der Effekt der Farbwahl angezeigt, und zwar als Umrandung und Feldhintergrund gegen Feldhintergrund und Umrandung in weiss und schwarz, sowie als Textfarbe auf grauem und weissem Hintergrund und als Hintergrund für schwarzen Text.

Die gewünschte Farbe wird mit Anklicken von **OK** übernommen.

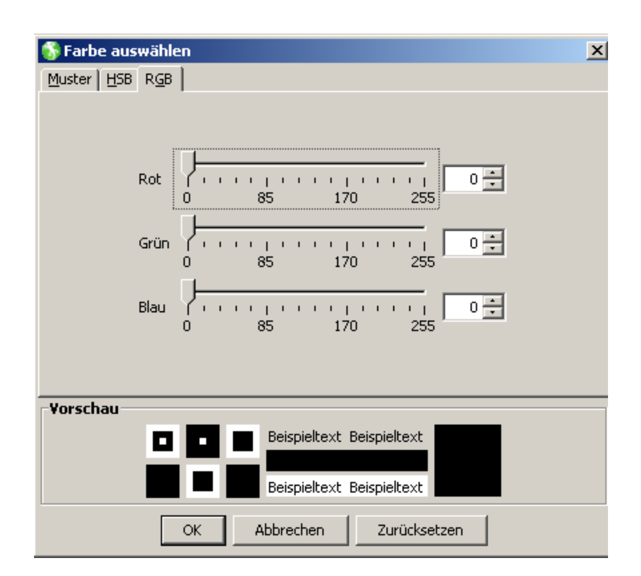
Fig. 3-30 *Farbauswahlfenster in HSB-Darstellung*



Bei Auswahl in der **HSB**-Darstellung wird der Farbton über den vertikalen Farbspektrumschieber in der Mitte des Fensters eingestellt. Der Helligkeitswert kann dann durch Anklicken des Musters im Farbquadrat links eingestellt werden. Gleichzeitig werden die aktuellen HSB-Werte und die resultierenden RGB-Werte in den Feldern rechts angezeigt.

HSB-Werke können zudem direkt in die Eingabefelder **H**, **S**, **B** eingetippt werden, oder mit Hilfe der **Auf-Ab**-Pfeilsymbole erhöht oder verringert werden. Der Wert für **H** liegt zwischen **0** und **360**, und für **S** und **B** liegt zwischen **0** und **100**.

Fig. 3-31 *Farbauswahlfenster in RGB-Darstellung*



In der **RGB**-Darstellung werden die Farben über drei Schieber für Rot, Grün und Blau eingestellt. Gleichzeitig werden im Anzeigefeld rechts des Schiebers die aktuellen Werte angezeigt.

Es ist auch möglich, die RGB-Werte direkt in die Felder einzutippen oder mit den **Auf-Ab**-Pfeilsymbolen zu erhöhen oder zu verringern. Der Wert von **R**, **G** und **B** liegt zwischen **0** und **255**.

3.3 Betriebsarten von Designer

Snapform Designer kennt zwei deutlich unterscheidbare Betriebsarten. Die Entwurfs-Ansicht ist die eigentlche Arbeitsumgebung für die Entwicklung von Formularen. Die Vorschau-Ansicht stellt das Formular so dar, wie es sich im Snapform Viewer zeigt. Gewisse Komponenten des Formulars werden erst in der Vorschau richtig dargestellt (z.B. HTML-formatierte Texte). Die Betriebsart wird mit Anklicken der betreffenden Schaltfläche in der Titelleiste des Editors umgeschaltet.

3.3.1 Entwurfs-Ansicht

Die Entwurfs-Ansicht ist die eigentliche Arbeitsumgebung für die Formularentwicklung. In der Entwurfs-Ansicht können Elemente auf der Arbeitsfläche platziert und deren Eigenschaften verändert werden.

In der Entwurfs-Ansicht sind die Werkzeugleiste und die Steuerelemente-Leiste aktiv. Ebenso sind sämtliche Menupunkte (soweit nicht anderweitig eingeschränkt) aktiv.

3.3.2 Vorschau

Die Vorschau dient zur Kontrolle des Aussehens des Formulars sowie zum Testen von Formeln und dem Verhalten bei Eingaben.

Wenn wieder in die Entwurfs-Ansicht umgeschaltet wird, werden sämtliche eingegebenen Werte verworfen und die Felder zurückgesetzt.

4 Formular-Entwicklung

4.1 Einleitung

Die Formularentwicklung ist die Kernaufgabe des Snapform Designers. In diesem Kapitel werden die verschiedenen Phasen der Formular-Entwicklung (Analyse, Grundlayout, Aktive Elemente, Logik, Sicherung, Verteilung und Verwaltung) mit SnapForm Designer beschrieben.

Dieses Kapitel ist prozessorientiert. Es behandelt die Abfolgen zur Erstellung funktionierender Formulare. Dabei werden verschiedene Elemente von Designer verwendet. Die für Aktionen und Logik verwendete Scriptsprache ist im Kapitel 5, "Formelsprache-Referenz" detailliert beschrieben.

Die Formularentwicklung umfasst im Wesentlichen folgende Phasen:

1. Analyse und Planung
2. Erstellen des Grundlayouts
3. Hinzufügen der Ein- und Ausgabefelder sowie der aktiven Elemente
4. Hinzufügen von Aktionen und Business-Logik
5. Aufbereiten zur Verteilung
6. Verteilung
7. Wartung

Diese Punkte werden in diesem Kapitel in eigenen Abschnitten behandelt.

4.2 Analyse und Planung

Wie bei allen Entwicklungsprojekten haben Analyse und Planung einen grossen Einfluss auf den Erfolg des Projekts. Auch wenn diese Phase nicht direkt mit der Benutzung von SnapForm Designer zusammenhängt, ist sie doch derart wichtig, dass ein paar Anmerkungen notwendig sind.

Im Rahmen der Analyse sind folgende Fragen zu stellen (und zu beantworten):

- Ist der unterliegende Geschäftsprozess bereits vollständig bekannt?
- Ist das Formular der einzige Träger des unterliegenden Geschäftsprozesses?
- Welche Informationen werden benötigt (Input)
- Wann (im Prozessablauf) werden die Informationen benötigt?
- Welche Informationen müssen angezeigt werden (Output)
- Wann im Prozessablauf müssen diese Informationen angezeigt werden?
- Was ist das Ausgabemedium für das Formular (Bildschirm, Bürodrucker, Druckkopierer, Druckerei)?
- Welche interaktiven Elemente sind benötigt?
- Welche Plausibilisierungen sind benötigt?
- Welche weitergehenden Berechnungen sind benötigt?
- Soll das Formular vorbefüllt werden?
- Sollen eingegebene Daten an ein Back-End-System übermittelt werden und wenn ja, in welcher Form?
- Gibt es Layout-Vorgaben oder Styleguides?

- Muss das Formular barrierefrei sein?
- Soll das Formular vom gleichen Benutzer für verschiedene Anwendungen ausgefüllt werden?
- Soll das Formular rein elektronisch eingesetzt werden, oder soll es auch leer ausgedruckt und von Hand ausgefüllt werden können?
- Hat es repetitive Elemente?

Basierend auf diesen Antworten kann nun das Formular aufgebaut werden.

Im Rahmen der Planung gilt es auch die verschiedenen “Highlight”-Farben festzulegen. Dies erfolgt über Stilvorlagen, welche im Abschnitt 4.7 beschrieben sind.

Hinweis: *Stilvorlagen sollten sorgfältig und mit Bedacht geplant werden. Mit gut durchdachten Stilvorlagen kann sich der Arbeitsaufwand für weitere Formulare stark verringern, und es wird ein einheitliches Erscheinungsbild sichergestellt.*

4.3 Grundlayout

Der erste Schritt bei der Entwicklung eines neuen Formulars ist die Erstellung des Grundlayouts. Das Grundlayout kann entweder als Hintergrundbild importiert werden (und ist in SnapForm Designer nicht weiter editierbar), oder es ist Bestandteil des eigentlichen Formulars (und in SnapForm Designer vollständig editierbar). Mischformen sind zudem beliebig möglich.

In vielen Fällen besteht bereits eine Print-Version des Formulars, welche auch als PDF-Dokument vorliegt. In diesem Fall kann dieses PDF in ein Hintergrundbild umgewandelt werden. Dieses Methode hat den Vorteil, dass eine praktisch vollständige Übereinstimmung mit Print-Formularen erreicht werden kann (einschliesslich hochwertiger Miktrotypografie), und andererseits bestehendes Material nicht neu entwickelt werden muss.

Wenn keine Vorlage besteht, kann das Grundlayout, ausgehend von einer leeren Seite in SnapForm Designer entwickelt werden.

4.3.1 Hintergrundbilder

Für Hintergrundbilder wird das **.hdt**-Format eingesetzt. Hintergrundbilder werden durch Konversion eines PDF-Dokuments erzeugt. Für die Konversion gibt es zwei Möglichkeiten:

1. Der Snapform PDF Converter
2. Der kostenfreie Konversions-Service von Ringler Informatik.

4.3.1.1 Snapform PDF Converter

Der Snapform PDF Converter ist im nachfolgenden Abschnitt beschrieben.

4.3.1.2 Konversions-Service

Der Konversions-Service von Ringler Informatik steht allen registrierten Benutzern von SnapForm Designer kostenfrei zur Verfügung. Im PDF-Format vorliegende Dokumente sind dazu an Ringler Informatik zu senden. Die konvertierten Dokumente werden normalerweise innerhalb 1 Arbeitstag zurückgesandt.

Hinweis: *Die an den Konversions-Service gesandten Dokumente werden nach erfolgreich durchgeführter Konversion bei Ringler Informatik gelöscht. Die konvertierten Dokumente werden nach einer gewissen Zeit (10 Tage) ebenso gelöscht. Diese Dokumente werden auf keine Art und Weise Dritten zugänglich gemacht.*

4.3.2 Der Snapform PDF Converter

Der Snapform PDF Converter ist eine eigenständige Applikation, welche PDF-Dokumente in das Snapform-Hintergrundformat (.hdt) umwandelt. Der Snapform PDF Converter kann einzelne Dokumente umwandeln oder als Batchverarbeitung alle Dokumente in einem gegebenen Ordner.

Der Snapform PDF Converter ist eine auf .NET basierende Windows-Applikation. Die Installation ist in Abschnitt 2.3.3 beschrieben.

4.3.2.1 Vorbereitung der PDF-Dokumente

Snapform PDF Converter kann alle PDF Versionen bis 1.6 (entsprechend Acrobat 7) verarbeiten. Die meisten Dokumente können ohne weitere Vorarbeiten konvertiert werden.

Es ist jedoch sinnvoll, die PDF-Dokumente vor der Konversion aufzubereiten.

Die Aufbereitung erfolgt in Acrobat Professional (Acrobat 6 und neuer) oder Acrobat Vollversion (Acrobat 5 und älter). Die nachstehenden Angaben beziehen sich auf Acrobat 7 Professional.

Als allererstes müssen allfällig vorhandene Sicherheitseinstellungen entfernt werden.

Die weitere Aufbereitung erfolgt mit dem PDF Optimizer (Menu **Advanced –> PDF Optimizer...**). Dabei sind folgende Einstellungen empfohlen:

1. Bilder auf 150 dpi (Farbe und Graustufen) bzw. 600 dpi (Schwarzweiss) herunterrechnen.
2. Gescannte Seiten nicht weiter optimieren
3. Schriftarten soweit wie möglich einbetten (dabei ist zu beachten, dass diese Schriftarten auf dem betreffenden Rechner installiert sein müssen).
4. Transparente Objekte nicht glätten
5. Objekte entfernen: Sämtliche Optionen selektieren
6. Aufräumen: Objekt-Kompression entfernen und sämtliche Optionen ausser Optimieren für schnelle Webansicht selektieren

Mit diesen Optionen wird erreicht, dass das Dokument für den Snapform PDF Converter bestmöglichst aufbereitet worden ist.

Hinweis: *Bei dynamischen Formularen sicherstellen, dass alle benötigten Formularelemente angezeigt werden und anschliessend das Dokument glätten. Hierzu den Debugger von Acrobat öffnen (***<Ctrl><J>** *bzw.* **<Cmd><J>***) und den folgenden Code eingeben*

```
this.flattenPages() ;
```

und mit **<Ctrl><Return>** *ausführen.*

Sind mehrere Dokumente zu konvertieren, können die aufbereiteten PDF-Dateien in ein gesondertes Verzeichnis verschoben werden. Das erlaubt dann den Snapform PDF Converter im Batchbetrieb zu nutzen.

Als weitere Möglichkeit zur Aufbereitung (insbesondere, wenn keine Acrobat Professional Version zur Verfügung steht (sondern nur Acrobat Standard oder Acrobat Elements), ist ein erneutes “Ausdrucken” mit Acrobat Distiller. Dieser “refrying” genannte Prozess gilt normalerweise als verpönt, ist jedoch in diesem Fall eine gute Methode um das Dokument zu vereinfachen. Für die Konversion sind zudem die “zusätzlichen” Eigenschaften von PDF (Links, Felder, Struktur, etc.) nicht notwendig.

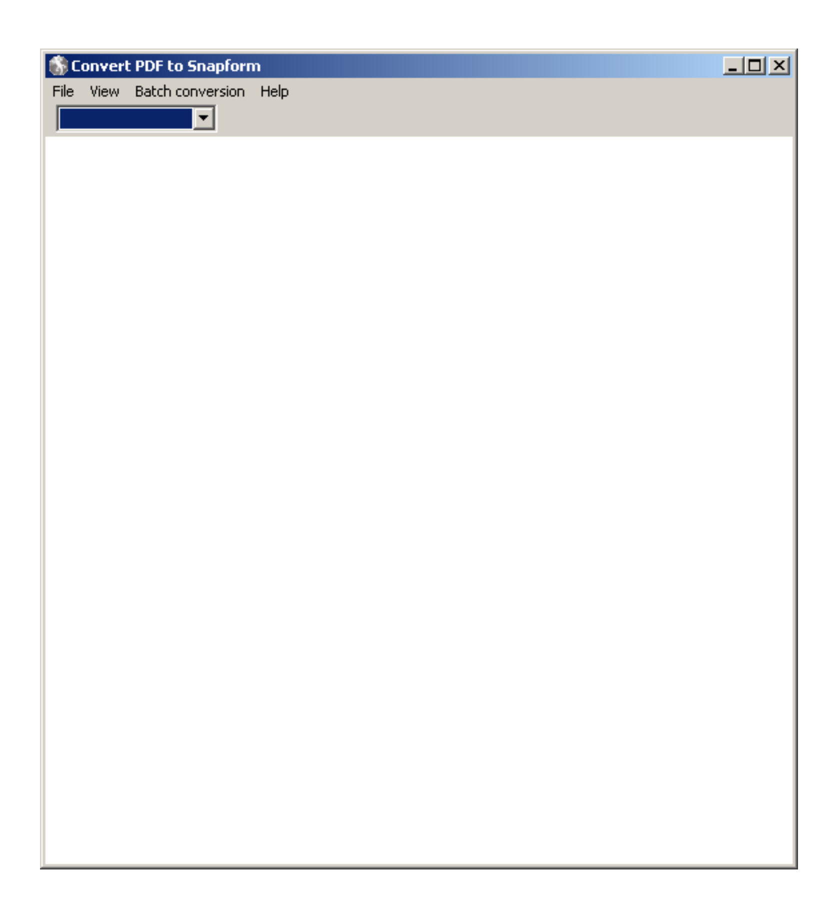
Bei den Optionen für Distiller ist darauf zu achten, dass die Schriften eingebettet werden (was zudem auch in der Einstellung des Druckertreibers kontrolliert werden sollte).

4.3.2.2 Die Benutzeroberfläche des Snapform PDF Converters

Der Snapform PDF Converter ist nur mit Benutzeroberfläche in englischer Sprache verfügbar.

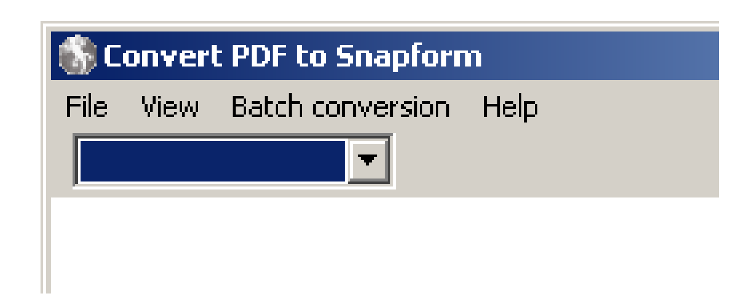
Nach dem Programmstart stellt sich der Snapform PDF Converter wie in Fig. 4-1 gezeigt dar.

Fig. 4-1 *Der Snapform PDF Converter*



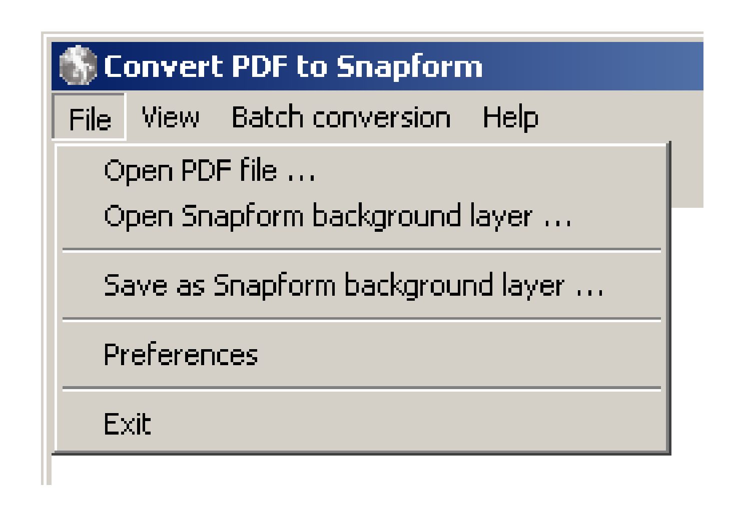
Die Menuleiste ist in Fig. 4-2 genauer dargestellt.

Fig. 4-2 *Die Snapform PDF Converter Menuleiste*



Die verschiedenen Menu-Elemente sind nachfolgend dargestellt.

Fig. 4-3 *Das* **File** *Menu des Snapform PDF Converters*



Das **File** Menu (siehe Fig. 4-3) umfasst die folgenden Menupunkte:

Open PDF file ...

Es öffnet sich ein normaler Datei-Öffnen Dialog zur Auswahl einer PDF-Datei. Die ausgewählte Datei wird konvertiert und das Resultat im Vorschau-Fenster angezeigt.

OpenSnapform background layer ...

Es öffnet sich ein normaler Datei-Öffnen Dialog zur Auswahl eines Snapform Hintergrundbildes (.hdt-Datei). Die ausgewählte Datei wird im Vorschau-Fenster angezeigt.

Save as Snapform background layer ...

Es öffnet sich ein normaler Datei-Speichern Dialog zur Auswahl des Speicherortes für das aktuelle Snapform Hintergrundbild. Das Hintergrundbild wird im .hdt-Format gespeichert.

Preferences

Es öffnet sich ein Dialog für Einstellungen zur Optimierung der Konversion.

Exit

Das Programm wird beendet.

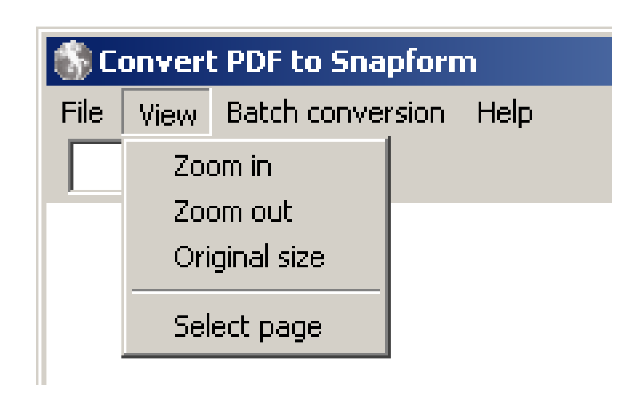


Fig. 4-4 *Das* ***View*** *Menu des Snapform PDF Converters*

Das **View** Menu (siehe Fig. 4-4)dient zur Kontrolle der Anzeige im Vorschau-Fenster. Es umfasst die folgenden Menupunkte:

Zoom in

Die Ansicht wird bei Klicken im Vorschau-Fenster um rund 10% vergrössert dargestellt.

Zoom out

Die Ansicht wird bei Klicken im Vorschau-Fenster um rund 10% verkleinert dargestellt.

Original size

Die Ansicht im Vorschau-Fenster wird in Originalgrösse (100%) dargestellt.

Select page

Wenn ein mehrseitiges Dokument geöffnet ist, erscheint ein Untermenu mit der Seitenauswahl (siehe Fig. 4-5). Die aktuelle Seite ist markiert.

Fig. 4-5 *Das* **Select page** *Untermenu des Snapform PDF Converters*

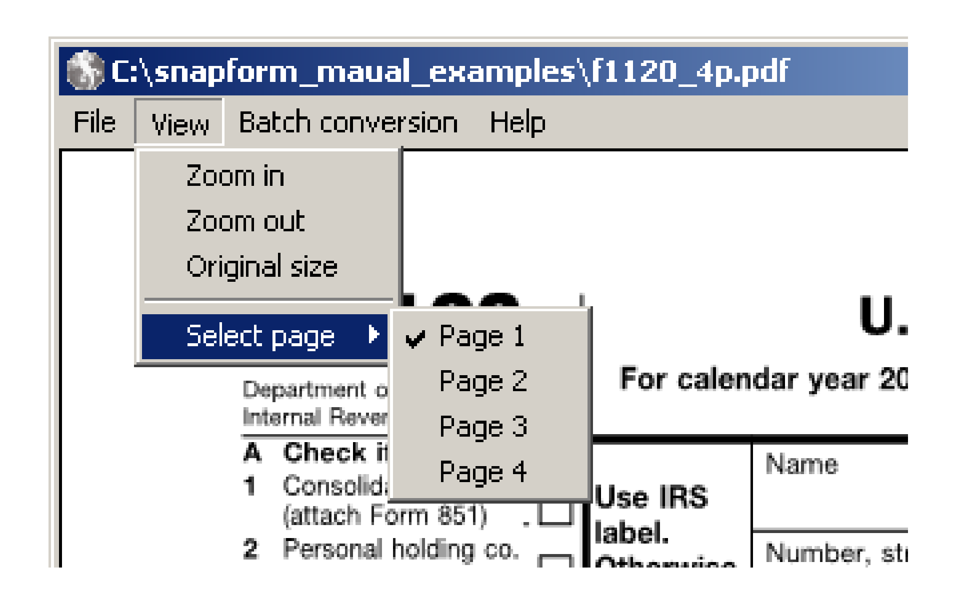
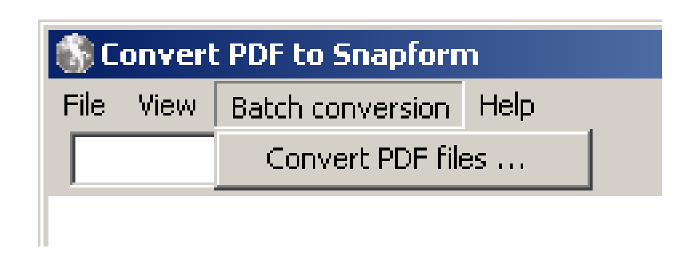


Fig. 4-6 *Das* **Batch conversion** *Menu des Snapform PDF Converters*



Das **Batch conversion** Menu (siehe Fig. 4-6) umfasst den einen Menupunkt **Convert PDF files...**, welcher den Dialog zur Auswahl des batchweise zu konvertierenden Ordners öffnet (siehe auch Abschnitt 4.3.2.5).

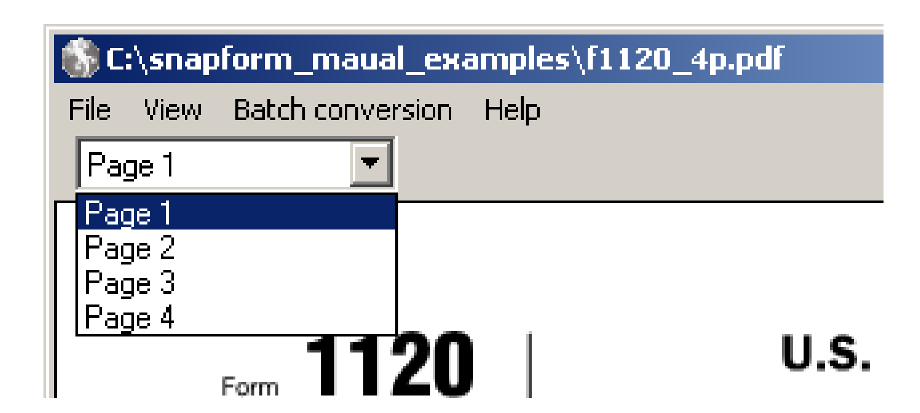
Fig. 4-7 *Das* **Help** *Menu des Snapform PDF Converters*



Das **Help** Menu (siehe Fig. 4-7) umfasst den einen Menupunkt **About**, welcher ein Fenster mit der Programmidentifikation öffnet.

In Ergänzung zum Select Page Untermenu besteht in einer Werkzeugleiste unterhalb der Menuleiste ein Drop Down Menu, welches ebenfalls die Seitenauswahl erlaubt (siehe Fig. 4-8).

Fig. 4-8 *Seitenauswahl Drop Down Menu*



4.3.2.3 Grundeinstellungen

Die Konversion eines PDF-Dokuments nach einem Snapform Hintergrundbild läuft normalerweise ohne grosse Schwierigkeiten ab, besonders wenn das PDF-Dokument vorgängig aufbereitet worden ist. Es gibt jedoch gewisse Fälle, in welchen die automatische Konvertierungslogik ansteht, oder die Konversion selbst sehr lange dauert. In diesen Fällen können im Menu **File –> Preferences** (siehe Fig. 4-9). gewisse Parameter gesteuert werden.

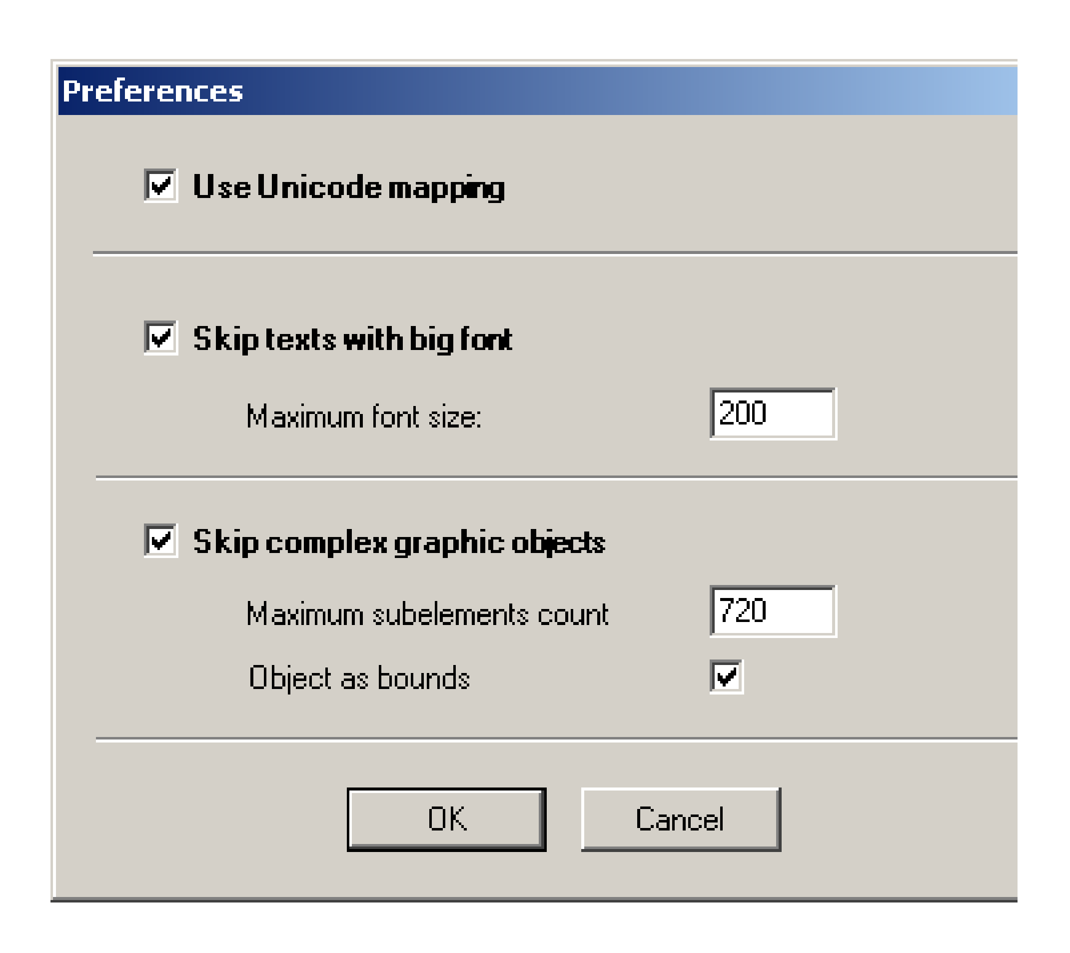


Fig. 4-9 *Der* **Preferences** *Dialog des Snapform PDF Converters*

Use Unicode mapping

Normalerweise wird in einem PDF-Dokument eine Schrift mit einer definierten Zeichensatztabelle (mapping table) eingebettet und verwendet. Es gibt jedoch Fälle, in denen diese Tabelle entweder fehlt, oder fehlerhaft ist, oder aber die auf dem Rechner installierte Version der betreffenden Schrift nicht exakt mit der im PDF-Dokument festgelegten Schrift übereinstimmt. Mit Wahl dieses Parameters wird zwangsweise die Zeichensatzcodierung gemäss Unicode benutzt.

Skip texts with big font

Es kann der Fall auftreten, dass (zum Beispiel bei Wasserzeichen) die verwendete Schrift sehr gross ist. Im Fall eines Wasserzeichens ist es jedoch eher nicht erwünscht, dass dieses konvertiert wird.

Dieser Parameter unterdrückt die Konversion von Text in einer Schriftgrösse grösser als im Wert **Maximum font size** angegeben. Der Grundwert beträgt 200 Points.

Skip complex graphic objects

Je nachdem, wie das PDF-Dokument ursprünglich erstellt worden ist, können sehr komplexe Vektorpfade enthalten sein. Die Umwandlung dieser Pfade kann einen beträchtlichen Rechenaufwand zur Folge haben, und auch die resultierende Dateigrösse stark steigern.

Dieser Parameter erlaubt es nun, komplexe Grafikelemente mit einer Anzahl Nodes über dem eingegebenen Schwellwert zu unterdrücken.

Der Schwellwert (**Maximum subelements count**) hat den Grundwert 750 Nodes.

Mit Wahl von **Objects as bounds** kann zudem auch die Anzeige des Beschneidungspfades des Grafikobjekts unterdrückt werden. Wenn diese Option gewählt ist, wird das Grafikobjekt auf keine Art und Weise konvertiert.

4.3.2.4 Einzeldokument konvertieren

Nachdem das PDF-Dokument gegebenenfalls aufbereitet worden ist, wird es im Snapform PDF Converter mit dem Menu **File –> Open PDF file…** geöffnet. Nach ein paar Momenten erscheint das konvertierte Dokument im Vorschau-Fenster, und kann kontrolliert und gespeichert werden.

4.3.2.5 Batch-Konversion

Sollen mehrere PDF-Dokumente konvertiert werden, empfiehlt sich die Benutzung der Batch-Konversion.

Die zu konvertierenden, gegebenenfalls aufbereiteten Dokumente werden in einem Verzeichnis gesammelt. Mit von **Batch conversion –> Convert PDF files…** öffnet sich der Dialog zur Batchverarbeitung (siehe Fig. 4-10).

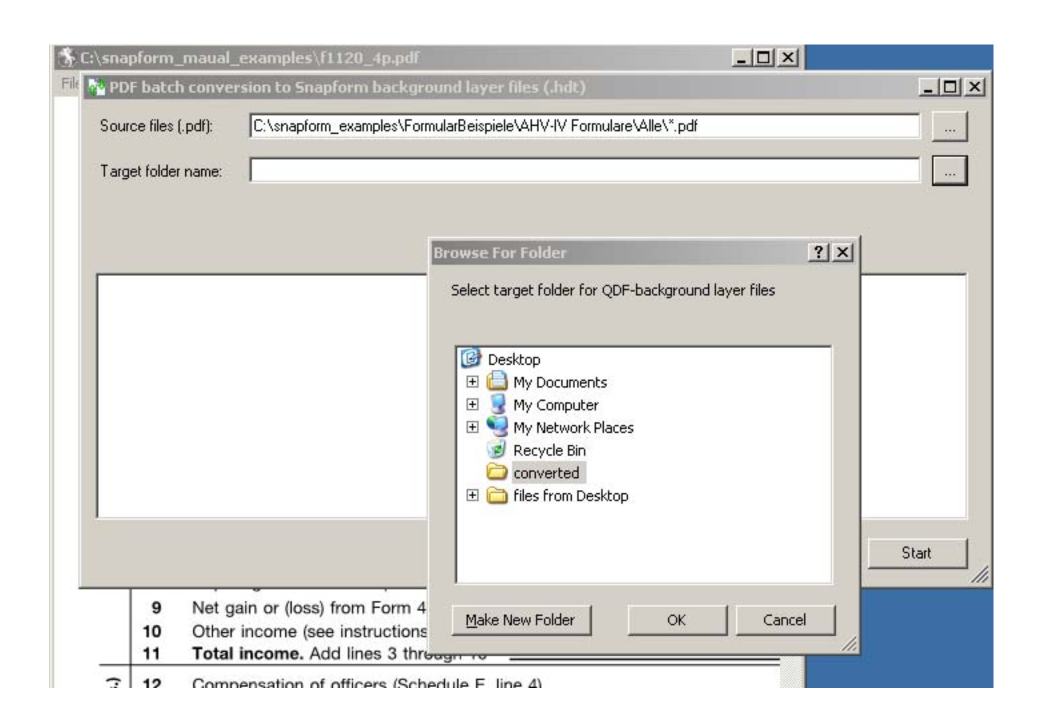
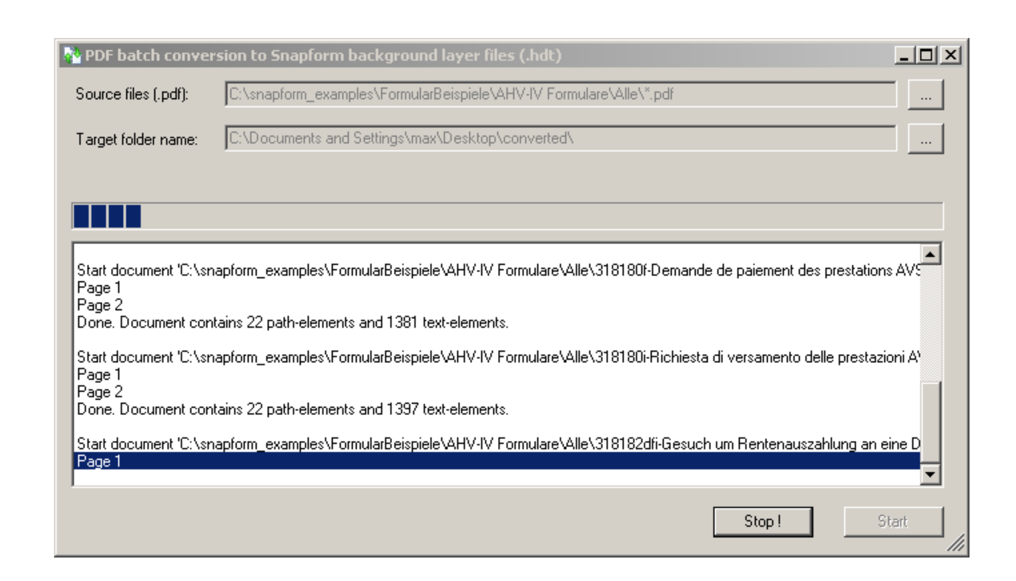


Fig. 4-10 *Batch-Konversion Fenster mit offenem Verzeichnis Wählen Dialog*

In der ersten Zeile **Source files (.pdf):** wird der Pfad zum Quellverzeichnis der Ursprungs-PDF-Dokumente angegeben. Anklicken der Schaltfläche **...** öffnet einen Verzeichnis Wählen-Dialog, welcher die Auswahl des betreffenden Verzeichnisses erlaubt. Analog erfolgt auch die Eingabe des Zielverzeichnisses in der Zeile **Target folder name:**.

Sind beide Verzeichnisse gewählt, wird der Konvertierungsprozess mit Anklicken der Schaltfläche **Start** ausgelöst. Nun werden die im Quellverzeichnis gefundenen PDF-Dokumente nacheinander konvertiert. Der Arbeitsfortschritt wird durch einen Balken dargestellt. Im Anzeigefenster wird zudem die Konversion für jedes Dokument protokolliert. Dabei wird neben dem Dateinamen auch ein Eintrag für jede konvertierte Seite erzeugt, und die Anzahl Pfad- und Textelemente wird angegeben (siehe Fig. 4-11).

Fig. 4-11 *Arbeitsfortschritt in der Batch-Konversion.*



Es muss sowohl das Quell- wie auch das Zielverzeichnis festgelegt worden sein. Wenn ein Eintrag fehlt, erscheint nach dem Anklicken der Schaltfläche **Start** eine Meldung **Source files are not specified** bzw. **Target folder is not specified**.

Die Batch-Konversion kann jederzeit mit Anklicken der Schaltfläche **Stop!** abgebrochen werden.

Wenn die Schaltfläche **Close** angezeigt ist, kann damit der Batch-Konversion Dialog geschlossen werden.

4.3.2.6 Probleme bei der Konversion

Es ist immer möglich, dass eine Konversion nicht durchgeführt werden kann. Wenn dies der Fall ist, sollte das betreffende PDF-Dokument in Acrobat Professional geöffnet und Seite für Seite durchgeblättert werden. Dabei ist auf Meldungen von Acrobat zu achten. In vielen Fällen reicht ein erneutes Abspeichern des Dokuments aus. Gegebenenfalls kann auch versucht werden, das Dokument noch einmal durch den Acrobat Distiller zu schicken (siehe Abschnitt 4.3.2.1).

Scheitert die Konversion immer auf einer bestimmten Seite, kann versucht werden, diese Seite aus dem Dokument zu extrahieren und separat zu konvertieren, und andererseits diese Seite aus dem

Dokument zu entfernen und das verbleibende Dokument zu konvertieren. Die beiden Teile können dann im SnapForm Designer mit der **Seiten hinzufügen**-Funktion wieder zusammengefügt werden.

Bedingt durch die unterschiedlichen internen Datenformate in PDF und dem Snapform Hintergrundbild kann es sein, dass gewisse Stilelemente nicht vollständig konvertiert werden. Es ist dann zu entscheiden, wieweit diese Merkmale wesentlich sind (und gegebenenfalls im PDF aufbereitet werden müssten), oder ob das Resultat der Konvertierung akzeptiert werden kann.

Bei der Batch-Konvertierung kann der Fall eintreten, dass der Konverter bei einer bestimmten Datei stecken bleibt, und auch ein Anklicken der Schaltfläche **Stop!** nicht genügt, um die Batch-Verarbeitung zu stoppen. In diesem Fall kann das Fenster mit Anklicken der Schliessmarke geschlossen werden; dies stoppt den Konvertierungsvorgang zuverlässig. Die Datei, welche den Stop verursacht muss nun entweder weitergehend aufbereitet, oder aus dem Quellverzeichnis entfernt werden.

4.3.3 Neues Dokument erstellen

Die Konversion eines PDF-Dokuments in ein Hintergrundbild ist im Wesentlichen eine Vorbereitung zum Erstellen eines neuen Dokuments, kann aber gänzlich unabhängig durchgeführt werden.

Der eigentliche Beginn der Arbeit mit SnapForm Designer beginnt nun mit dem Erstellen eines neuen Dokuments. Neue Dokumente können entweder als leere Seiten oder durch Import von bestehenden Dokumenten erstellt werden.

Mit dem Menu **Datei –> Neues Formular…** öffnet sich der **Neues Format erstellen** Dialog (siehe Fig. 4-12).

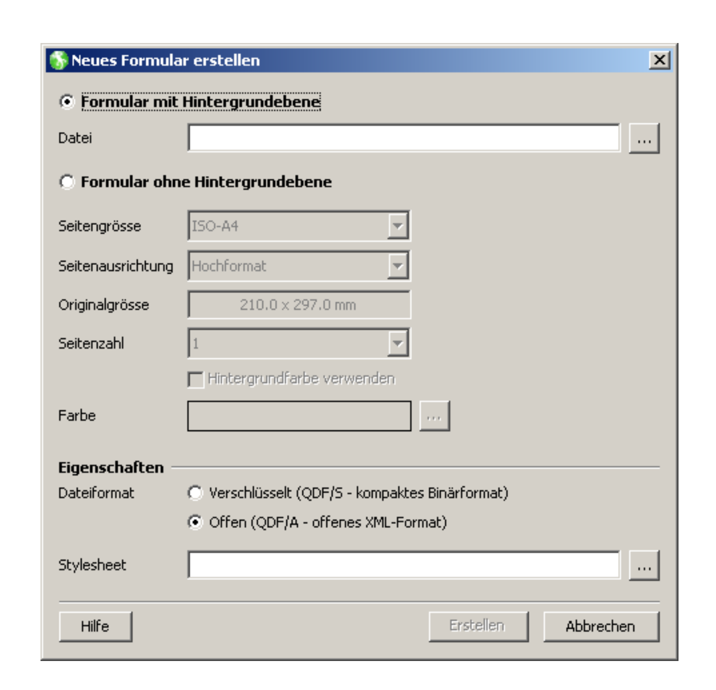


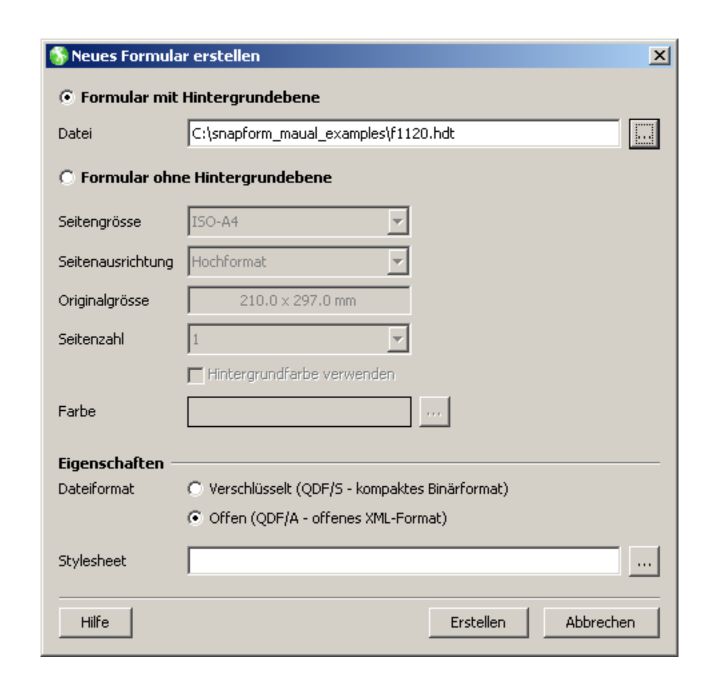
Fig. 4-12 *Der* **Neues Format erstellen** *Dialog*

Der Neues Format erstellen Dialog besteht aus drei Zonen: **Formular mit Hintergrundebene**, **Formular ohne Hintergrundebene** sowie **Eigenschaften**. Letztere Zone gilt gleichermassen für die übrigen Optionen.

4.3.3.1 Import bestehender Hintergrundseiten

Der schnellste Weg, ein Formular zu erstellen bietet sich an, wenn bereits ein Hintergrundbild in Form einer .hdt Datei vorhanden ist. Wenn die Option **Formular mit Hintergrundebene** gewählt wird, steht eine Eingabezeile für den Pfad zur .hdt Datei zur Verfügung. Die Schaltfläche **...** öffnet zudem einen normalen Datei Öffnen-Dialog, welcher die Wahl der Datei erlaubt (siehe Fig. 4-13).

Fig. 4-13 *Neues Formular mit Hintergrundebene.*



Die resultierende Seitengrösse richtet sich nach den Abmessungen des Hintergrundbildes. Die Anzahl Seiten entspricht ebenfalls der Seitenzahl des Hintergrundbildes.

Nach Wahl der .hdt Datei sind verschiedene Eigenschaften des Formulars zu wählen, damit dieses anschliessend erzeugt werden (siehe Abschnitt 4.3.3.3).

Hinweis: *Bei einem auf diese Art und Weise erzeugtem Formular kann das Hintergrundbild nicht in Snapform Designer editiert werden. Wenn das Hintergrundbild geändert werden muss, muss auf das ursprüngliche Originaldokument, aus welchem das PDF erzeugt wurde, zurückgegriffen und die Änderungen in diesem Dokument vorgenommen werden.*

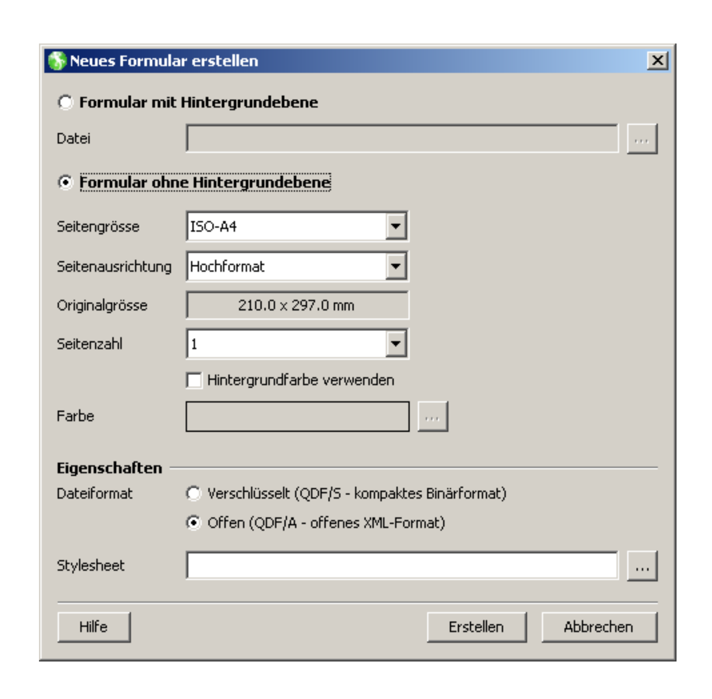
4.3.3.2 Leere Seiten erzeugen

Besteht kein passendes Hintergrundbild im .hdt Format, oder ist zu erwarten, dass das Grundlayout in Zukunft häufig geändert wird, ist es besser, ein gänzlich leeres Dokument zu erzeugen.

Hinweis: *Es ist jederzeit möglich, in einem bestehenden Formular einen Hintergrund einzufügen (siehe Abschnitt 4.3.4.2).*

Wenn die Option Formular ohne Hintergrundebene gewählt wird, sind im Dialog **Neues Formular erstellen** 5 Parameter zugänglich (siehe Fig. 4-14).

Fig. 4-14 *Neues Formular ohne Hintergrundebene*



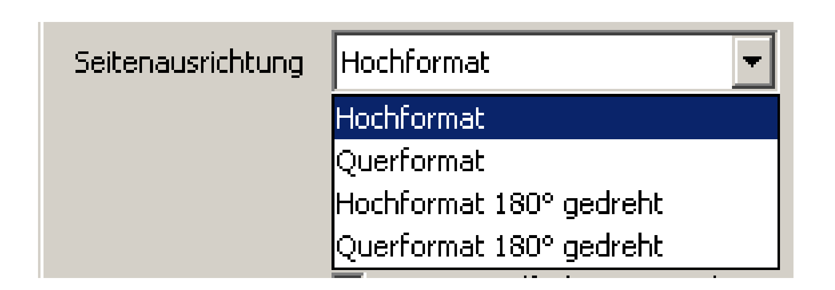
Seitengrösse

Dieses Drop Down Menu umfasst eine grosse Liste an Papierformaten aus der ganzen Welt.

Seitenausrichtung

Dieses Drop Down Menu umfasst eine Liste an möglichen Seitenausrichtungen (siehe Fig. 4-15)

Fig. 4-15 *Drop Down Menu Seitenausrichtung*



Die Optionen **Hochformat 180° gedreht** bzw. **Querformat 180° gedreht** sind für Anwendungen gedacht, bei denen beispielsweise Vorder- und Rückseite ausgedruckt wird.

Originalgrösse

Die Masse des gewählten Formats. Dieses Feld kann nicht verändert werden.

Seitenzahl

Anzahl zu generierender Seiten. Das Drop Down Menu erlaubt die Wahl von 1 bis 20 Seiten.

Sollten mehr Seiten benötigt werden, können weitere Seiten auf einfache Art und Weise hinzugefügt werden (siehe auch Abschnitt 4.3.4.1).

Hintergrundfarbe verwenden

Wenn dieses Kontrollkästchen gewählt ist, wird der Hintergrund des Formulars mit einer im Auswahlfeld **Farbe** gewählten Farbe versehen.

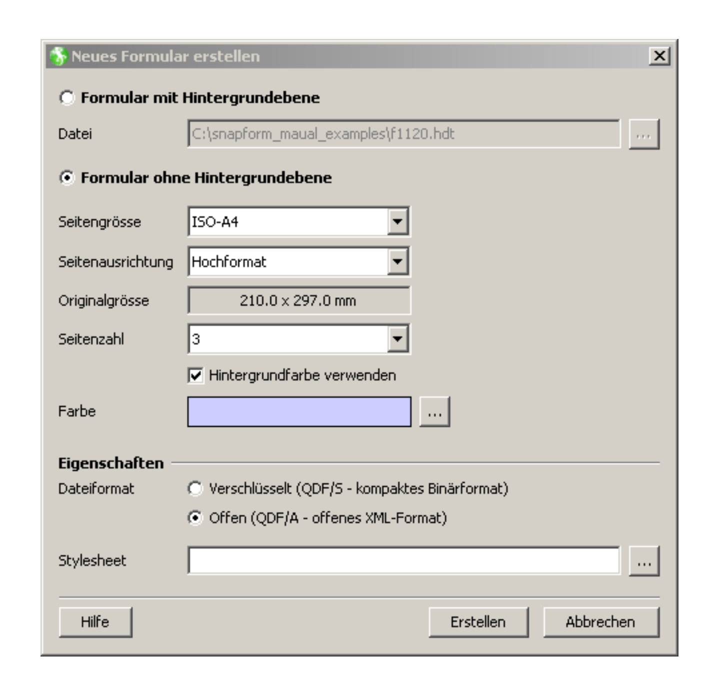
Farbe

Wenn das Kontrollkästchen **Hintergrundfarbe verwenden** angeklickt ist, ist dieses Feld aktiv. Mit Doppelklick in dieses Feld bzw. Anklicken der Schaltfläche **...** öffnet sich das Farbauswahlfenster (siehe Abschnitt 3.2.9.2), welches die Definition der gewünschten Farbe erlaubt.

4.3.3.3 Eigenschaften von neuen Formularen

Nachdem entweder das Hintergrundbild oder die Parameter für die Formulargrössen gewählt worden sind, sind noch zwei weitere Parameter notwendig, damit das Formular erzeugt werden kann. Diese Parameter sind in der Zone Eigenschaften zusammengefasst (siehe Fig. 4-16).

Fig. 4-16 *Eigenschaften des neuen Formulars*



Dateiformat

Das Snapform-Dateiformat kennt zwei Varianten: verschlüsselt/komprimiert und offen.

Mit der Wahl von **verschlüsselt (QDF/S)** wird die Datei beim Speichern vollständig verschlüsselt und komprimiert. Dies führt zur kleinstmöglichen Dateigrösse, bedingt aber eine Snapform-Applikation zum Auslesen der Daten.

Mit der Wahl von **offen (QDF/A)** werden die Nutzdaten der Datei im XML-Format offen gespeichert (Layout und Hintergrundebene werden in jedem Fall komprimiert). Dies führt zu grösseren Dateien. Dafür ist es möglich, die Nutzdaten auch ohne Snapform-Applikation auszulesen (z.B. durch ein Document Management System).

Welche Variante gewählt wird, hängt stark vom vorgesehenen Arbeitsablauf der Formulare ab und muss gegebenenfalls von Fall zu Fall entschieden werden.

Stylesheet

Es ist möglich, Eigenschaften von Feldern und anderen Formularelementen in einer Stilvorlage zu speichern. Dieses Stylesheet kann bei der Erzeugung des Formulars direkt eingebunden werden.

Damit werden die darin definierten Eigenschaften automatisch für alle eingefügten Elemente als Vorgabe gewählt.

Stylesheets sind im Abschnitt 4.7 genauer beschrieben.

Mit Anklicken der Schaltfläche **...** öffnet sich ein Datei Öffnen Dialog, welcher die Auswahl der betreffenden Datei ermöglicht.

Mit diesen Optionen ist das neue Formular definiert, und es kann mit Anklicken der Schaltfläche **Erstellen** erstellt werden.

4.3.4 Dokumentseiten

Formulare sind weitgehend seitenorientiert. Daher bietet auch Snapform Designer Funktionen, welche das Verwalten von Seiten in Formularen erlauben. Seiten können hinzugefügt und entfernt werden, und Seiten haben eine Reihe von Eigenschaften. In mehrseitigen Formularen kann jede Seite individuelle Eigenschaften aufweisen.

4.3.4.1 Seiten hinzufügen und entfernen

Zum Hinzufügen und Entfernen von Seiten stehen drei Befehle im Menu **Bearbeiten** zur Verfügung.

Neue Seite hinzufügen

Wird Menu **Edit – Neue Seite hinzufügen ...** gewählt, öffnet sich der **Seite hinzufügen** Dialog (siehe Fig. 4-17). Dieser Dialog ist demjenigen des Erstellen einer neuen Seite ohne Hintergrundbild sehr ähnlich.

Fig. 4-17 *Seite hinzufügen Dialog 029*



Seitengrösse

Dieses Drop Down Menu umfasst eine grosse Liste an Papierformaten aus der ganzen Welt.

Seitenausrichtung

Dieses Drop Down Menu umfasst eine Liste an möglichen Seitenausrichtungen (siehe Fig. 4-15)

Die Optionen **Hochformat 180° gedreht** bzw. **Querformat 180° gedreht** sind für Anwendungen gedacht, bei denen beispielsweise Vorder- und Rückseite ausgedruckt wird.

Originalgrösse

Die Masse des gewählten Formats. Dieses Feld kann nicht verändert werden.

Hintergrundfarbe verwenden

Wenn dieses Kontrollkästchen gewählt ist, wird der Hintergrund des Formulars mit einer im Auswahlfeld **Farbe** gewählten Farbe versehen.

Farbe

Wenn das Kontrollkästchen **Hintergrundfarbe verwenden** angeklickt ist, ist dieses Feld aktiv. Mit Doppelklick in dieses Feld bzw. Anklicken der Schaltfläche **...** öffnet sich das Farbauswahlfenster (siehe Abschnitt 3.2.9.2), welches die Definition der gewünschten Farbe erlaubt.

Mit Anklicken von **Erstellen** wird die Seite erzeugt und dem Dokument angefügt. Die neue Seite wird zudem zur aktuellen Seite.

Neue Seite einfügen

Das Menu **Edit –> Neue Seite einfügen …** funktioniert identisch, fügt die neue Seite jedoch vor der aktuellen Seite ein. Die neue Seite wird zudem zur aktuellen Seite.

Seite entfernen

Entfernt werden Seiten mit dem Menu **Edit –> Diese Seite löschen**. Dieser Befehl löscht die aktuelle Seite.

Vorsicht: *Dieser Befehl entfernt die aktuelle Seite ohne weitere Warnung und ohne Möglichkeit des Rückgängig machens.*

Hinweis: *Nach dem Löschen der aktuellen Seite wird wieder die erste Seite des Dokuments angezeigt.*

4.3.4.2 Seiteneigenschaften

Die Seiteneigenschaften werden im Eigenschaften-Fenster (siehe Fig. 4-18) angezeigt, wenn kein Element der aktuellen Seite gewählt ist. Seiteneigenschaften beziehen sich normalerweise auf die aktuelle Seite, sie können jedoch auch für das gesamte Formular gelten (siehe Beschreibungen).

Fig. 4-18 *Das Seiteneigenschaften Fenster 030*

Eigenschaften von Formularseite

Eigenschaft	Wert
Formularname	
Hintergrundfarbe drucken	☐
Doppelte Namen zulassen	☐
Prüfregel	
Seitenbreite	595.2756
Seitenhöhe	419.5276
Papierformat	ISO-A5
Seitenausrichtung	Querformat
Gegenüberliegende Seite	
sichtbar	☑
druckbar	☑
Hintergrundbild	

Die ersten drei Eigenschaften gelten jeweils für das gesamte Formular; die übrigen Eigenschaften nur für die aktuelle Seite.

Formularname

Erlaubt die Vergabe eines Formularnamens. Dieser Name wird als interne Referenz benutzt.

Hintergrundfarbe drucken

Wenn diese Option gewählt ist, wird die Hintergrundfarbe mit ausgedruckt. Wenn sie nicht gewählt ist, wird kein farbiger Seitenhintergrund ausgedruckt. Dies hat den Vorteil, dass sowohl den visuellen Bedürfnissen am Bildschirm, wie auch den Anforderungen beim Drucken Rechnung getragen werden kann.

Doppelte Namen zulassen

Mit dieser Option kann festgelegt werden, ob Felder mit gleichen Feldnamen zulässig sind. Wenn dies der Fall ist, können Felder beispielsweise über alle Seiten dupliziert werden, und beim Ausfüllen erhalten sie automatisch den selben Wert.

Prüfregel

Es öffnet sich der Formeleditor zur Eingabe einer Formel, welche beim Öffnen der Seite ausgeführt wird.

Seitenbreite

Breite der Seite in der Default-Masseinheit. Dieses Feld dient zur Information und kann nicht verändert werden.

Seitenhöhe

Höhe der Seite in der Default-Masseinheit. Dieses Feld dient zur Information und kann nicht verändert werden.

Papierformat

Beim Anwählen dieser Eigenschaft öffnet sich ein Drop Down Menu mit der Liste der Papierformate, wie dies auch beim Einfügen von Seiten oder beim Definieren eines neuen Dokuments der Fall ist. Das gewählte Format wird sofort angewandt.

Seitenausrichtung

Beim Anwählen dieser Eigenschaft öffnet sich ein Drop Down Menu mit der Liste der Seitenausrichtungen, wie dies auch beim Einfügen von Seiten oder beim Definieren eines neuen Dokuments der Fall ist.

Gegenüberliegende Seite

Beim Anwählen dieser Eigenschaft öffnet sich ein Drop Down Menu mit der Auswahl der Seiten in diesem Dokument. Diese Eigenschaft wird für die Paginierung des fertigen Dokuments genutzt.

sichtbar

Mit diesem Kontrollkästchen wird eingestellt, ob die aktuelle Seite sichtbar sein soll.

druckbar

Mit diesem Kontrollkästchen wird eingestellt, ob die aktuelle Seite druckbar sein soll.

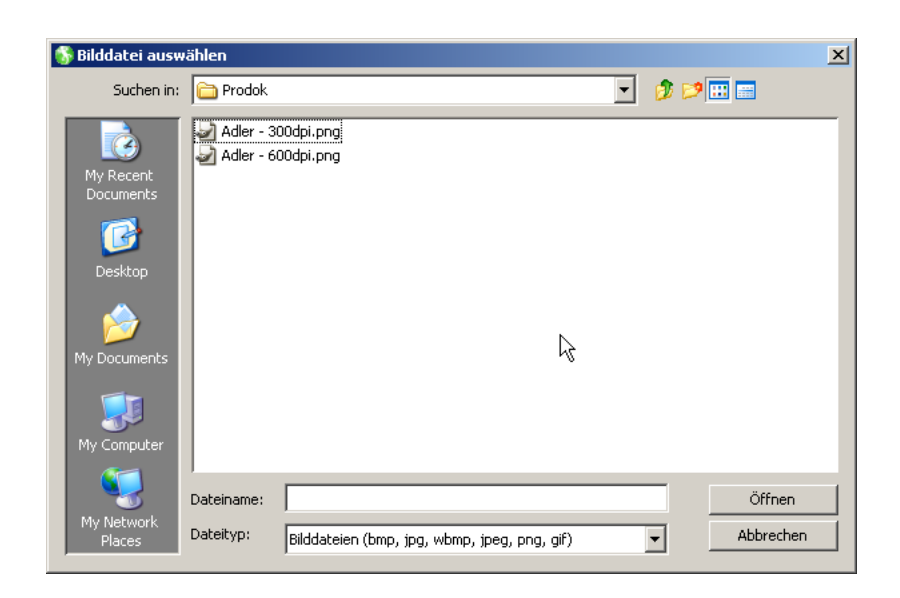
Hintergrundbild

Beim Anwählen dieser Eigenschaft kann der Pfad zu einer Grafikdatei angegeben werden, welche als Hintergrundbild eingefügt werden soll. Sind bereits Bilder eingefügt worden, sind deren Namen in einem Drop Down Menu aufgeführt.

Das oberste Element des Drop Down Menus ist leer, und entspricht "keinem" Bild. Mit Wahl dieses Elements kann ein eingefügtes Hintergrundbild entfernt werden.

Die Schaltfläche **...** öffnet einen Datei Öffnen Dialog, welcher die Wahl der betreffenden Datei erlaubt (siehe Fig. 4-19).

Fig. 4-19 *Datei Öffnen Dialog für Hintergrundbild 031*



Als Hintergrundbild können Grafikdateien der Formate JPEG, BMP, TIFF, GIF und PNG geladen werden. Andere Grafikformate, insbesondere Vektorformate, müssen in einem vorgängigen Schritt konvertiert bzw. gerendert werden.

Die importierte Hintergrunddatei wird seitenfüllen skaliert, was bedeutet, dass die Proportionen nicht erhalten bleiben.

Praxistip: *Besonders wenn das Hintergrundbild mit gedruckt werden soll, ist zu beachten, dass das JPEG-Format bei grafischen Bildern (z.B. Texten) Artefakte aufweisen kann, welche sich beim Ausdruck als Flecken darstellen. Grafische Bilder sind vorzugsweise im TIFF oder PNG-Format zu importieren.*

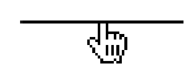
4.3.5 Grafikelemente

Snapform kennt vier Grafikelemente: Linien, Rechtecke, (Abdeck-)Flächen und Bilder. Die Grafikelemente werden als Bestandteil des Grundlayouts des Formulars betrachtet. Sie sind, da Bestandteil der Formularstruktur, unabhängig von Hintergrundbildern.

4.3.5.1 Linien

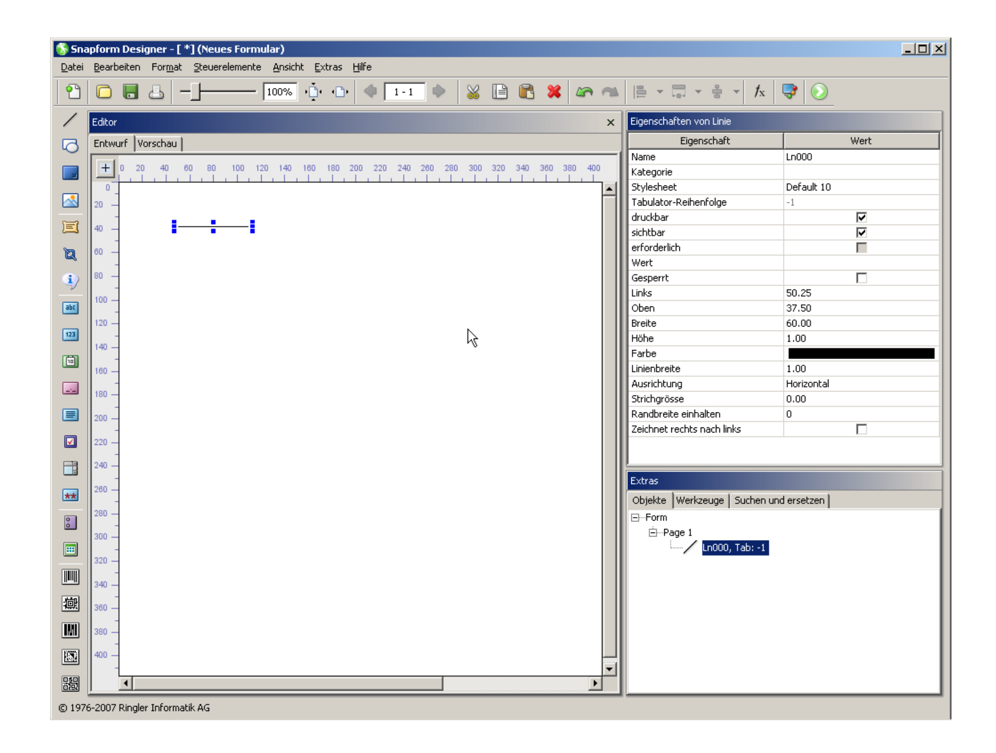
Das Linienwerkzeug erzeugt einfache gerade Linien. Nach Wahl des Werkzeugs erscheint ein Cursor mit einer horizontalen Linie (siehe Fig. 4-20).

Fig. 4-20 *Linienwerkzeug-Cursor*



Die mit dem Linienwerkzeug verknüpfte Linie ist eine schwarze horizontale Linie, 1 pt stark und 60 pt lang. Mit Mausklick wird diese Linie auf der Arbeitsfläche platziert. Dabei erhält sie automatisch einen Namen, bestehend aus den Zeichen Ln und einer dreistelligen fortlaufenden Nummer, beginnend mit 000. Die erste in einem Formular platzierte Linie trägt daher den Namen Ln000. In der Objektstruktur wird sie der betreffenden Seite hinzugefügt. In Fig. 4-21 ist die erste Linie des Formulars platziert worden, und anschliessend mit Anklicken selektiert.

Fig. 4-21 *Neu erstellte Linie 032*



Die Länge und Position der Linie kann durch Ziehen der Ankerpunkte des Selektionsrechteck verändert werden, die Position der Line auch durch "anfassen" und ziehen. Beim Überfahren der Ankerpunkte ändert der Cursor in Doppelpfeile, welche die Bewegungsrichtung des Ankerpunkts angeben. Beim "anfassen" und Drücken der Maustaste ändert sich der Cursor zu einem Doppelpfeilkreuz. Es ist zu beachten, dass die Ausrichtung der Linie sich auch dann nicht ändert, wenn das Selektionsrechteck quer zur Linie vergrössert wird; die Linie bleibt in der Mitte dieses Rechtecks zentriert.

Die übrigen Merkmale der Line werden über das Eigenschaften-Fenster verändert (siehe Fig. 4-22).

Fig. 4-22 *Eigenschaften einer Linie 032*

Eigenschaften von Linie

Eigenschaft	Wert
Name	Ln000
Kategorie	
Stylesheet	Default 10
Tabulator-Reihenfolge	-1
druckbar	☑
sichtbar	☑
erforderlich	☐
Wert	
Gesperrt	☐
Links	50.25
Oben	37.50
Breite	60.00
Höhe	1.00
Farbe	
Linienbreite	1.00
Ausrichtung	Horizontal
Strichgrösse	0.00
Randbreite einhalten	0
Zeichnet rechts nach links	☐

Name

Der Name der Linie im Dokument. Dieser Name kann geändert werden.

Kategorie

Zugehörigkeit zu einer oder mehreren Kategorien, welche es erlauben, logisch zusammengehörende Gruppen von Elementen zu bilden. Beim Anklicken dieses Felds öffnet sich der Kategorien Dialog (siehe Fig. 4-23).

Fig. 4-23 *Der Kategorien Dialog 033*

Eigenschaften von Linie

Eigenschaft	Wert
Name	Ln000
Kategorie	
Stylesheet	Default 10
Tabulator-Reihenfolge	-1
druckbar	☑
sichtbar	☑
erforderlich	☐
Wert	
Gesperrt	☐
Links	50.25
Oben	37.50
Breite	60.00
Höhe	1.00
Farbe	
Linienbreite	1.00
Ausrichtung	Horizontal
Strichgrösse	0.00
Randbreite einhalten	0
Zeichnet rechts nach links	☐

Die Kategorien werden durch Anklicken des betreffenden Kontrollkästchens ausgewählt. Mit **OK** wird die Auswahl übernommen.

Stylesheet

Das für das Element gültige Stylesheet. Wenn explizite kein Stylesheet ausgewählt worden ist, wird das erste Default-Stylesheet genutzt. Für Linien hat die Stylesheet-Definition allerdings keine Bedeutung.

Tabulator-Reihenfolge

Ordnungsnummer des Elements in der Tabulator-Reihenfolge. Dieser Wert ist immer -1 und kann nicht verändert werden. Das bedeutet, dass das Element nicht Bestandteil der Tabulatoren-Reihenfolge ist, was an sich logisch ist, da eine Linie nicht ein interaktives Element ist.

druckbar

Wenn das Kontrollkästchen gewählt ist, wird die Linie ausgedruckt.

sichtbar

Wenn das Kontrollkästchen gewählt ist, wird die Linie auf dem Bildschirm angezeigt.

erforderlich

Diese Eigenschaft ist immer ausgeschaltet, und kann nicht geändert werden.

Wert

Mit Anklicken dieser Eigenschaft öffnet sich der Formeleditor zur Eingabe einer Formel. Für Linienelemente wird eine Formel jedoch nur sehr selten benutzt.

Links

X-Koordinate der linken oberen Ecke der Umrandungsbox des Elements in Points. Diese Eigenschaft erlaubt eine genaue numerische Positionierung der Linie.

Hinweis: *Für eine genaue Positionierung der Linienmitte muss gegebenenfalls die halbe Linienbreite mit berücksichtigt werden.*

Oben

Y-Koordinate der linken oberen Ecke der Umrandungsbox des Elements in Points. Diese Eigenschaft erlaubt eine genaue numerische Positionierung der Linie.

Hinweis: *Für eine genaue Positionierung der Linienmitte muss gegebenenfalls die halbe Linienbreite mit berücksichtigt werden.*

Breite

Breite der Umrandungsbox der Linie. Bei einer horizontalen Linie entspricht dies der Länge der Linie; bei einer vertikalen Linie deren Linienstärke.

Höhe

Höhe der Umrandungsbox der Linie. Bei einer horizontalen Linie entspricht dies der Linienstärke; bei einer vertikalen Linie deren Länge.

Farbe

Farbe der Linie. Mit Anklicken der Eigenschaft öffnet sich das Farbauswahlfenster (siehe Abschnitt 3.2.9.2), welches die Definition der gewünschten Farbe erlaubt.

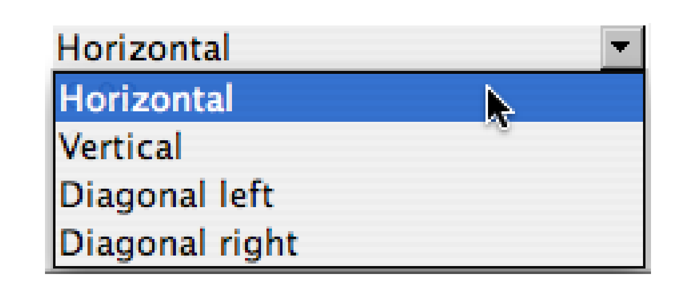
Linienbreite

Breite der Linie in Points.

Ausrichtung

Bei Wahl dieser Eigenschaft öffnet sich ein Drop Down Menu (siehe Fig. 4-24).

Fig. 4-24 *Ausrichtung Drop Down Menu in Linien-Eigenschaften 034*



Die Auswahlmöglichkeiten haben folgende Bedeutung:

Horizontal

Die Linie verläuft horizontal von links nach rechts.

Vertikal

Die Linie verläuft vertikal von oben nach unten.

Diagonal links

Die Linie verläuft diagonal in der Umrandungsbox von der linken unteren Ecke zur rechten oberen Ecke.

Diagonal rechts

Die Linie verläuft diagonal in der Umrandungsbox von der linken oberen Ecke zur rechten unteren Ecke.

Strichgrösse

Dieser Wert bestimmt, ob die Linie gestrichelt oder ganz dargestellt ist. Der Wert 0 bedeutet, dass die Linie ganz dargestellt wird. Bei anderen

Werten wird die Linie gestrichelt dargestellt, wobei der Wert die Strichlänge ausmacht.

Praxistip: *Eine punktierte Linie kann simuliert werden, indem die Strichgrösse der Linienbreite gleichgesetzt wird.*

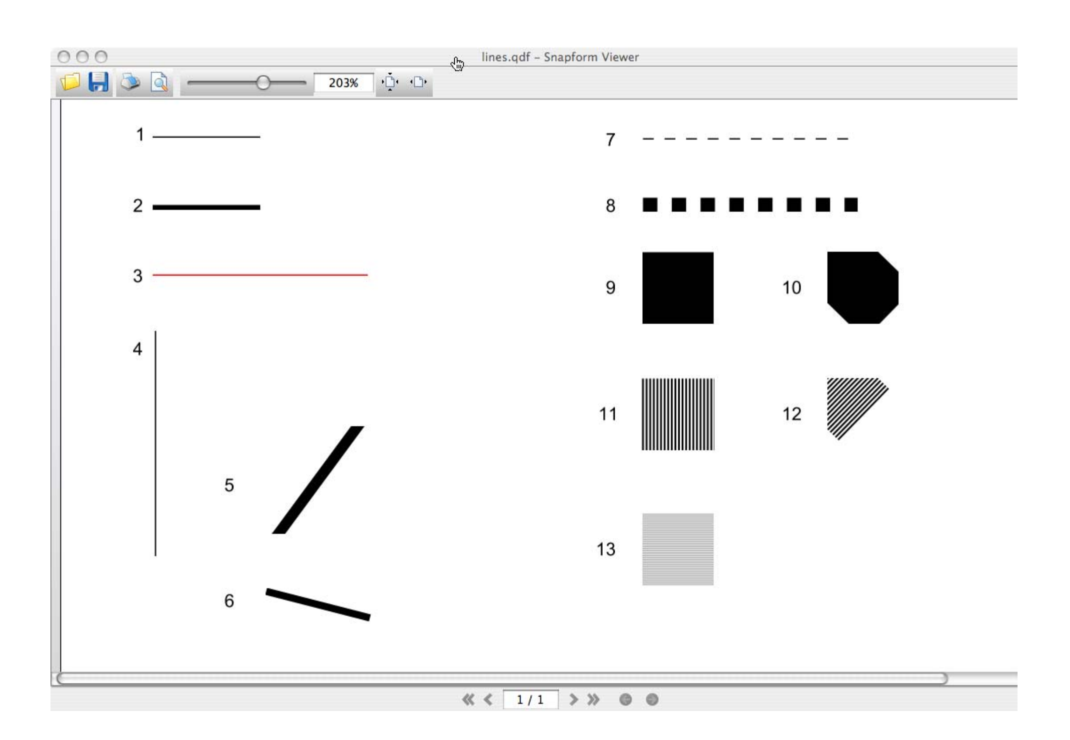
Randbreite einhalten

// Bedeutung unklar...

Praxistip: *Am einfachsten für die Arbeit ist es, die mit dem Lininenwerkzeug-Cursor verknüpfte Linie mit Mausklick in der Nähe des gewünschten Orts zu platzieren, und dann über.das Eigenschaften-Fenster vollständig zu definieren.*

Im Dokument linien.qdf (siehe Fig. 4-25) sind verschiedene Beispiele von Linien gezeigt. Nach dem Selektieren der Linien gibt das Eigenschaften-Fenster Auskunft über die Merkmale der jeweiligen Linie.

Fig. 4-25 *Linienbeispiele von linien.qdf*



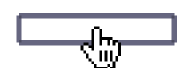
1 Ln000: Default Linie: horizontal, 60 pt lang, 1 pt stark, schwarz
2 Ln001: horizontal, 60 pt lang, 3 pt stark, schwarz
3 Ln002: horizontal, 120 pt lang, 1 pt stark, rot
4 Ln003: vertikal, 125 pt lang, 1 pt stark, schwarz

5 Ln004: diagonal links, Umrandungsbox 60 x 60 pt, 6 pt stark, schwarz
6 Ln005: diagonal rechts, Umrandungsbox 60 x 20 pt, 4 pt stark schwarz
7 Ln006: horizontal, 120 pt lang, 1 pt stark, Strichgrösse 6 pt, schwarz
8 Ln007: horizontal, 120 pt lang, 8 pt stark, Strichgrösse 8 pt, schwarz
9 Ln008: horizontal, 40 pt lang, 40 pt stark, schwarz
10 Ln009: diagonal rechts, Umrandungsbox 40 x 40 pt, 40 pt stark, schwarz
11 Ln010: horizontal, 40 pt lang, 40 pt stark, Strichgrösse 1 pt, schwarz
12 Ln011: diagonal rechts, Umrandungsbox 40 x 40 pt, 40 pt stark, Strichgrösse 1 pt, schwarz
13 Ln012: vertikal, 40 pt lang, 40 pt stark, Strichgrösse 0.5 pt, grau

4.3.5.2 Rechtecke

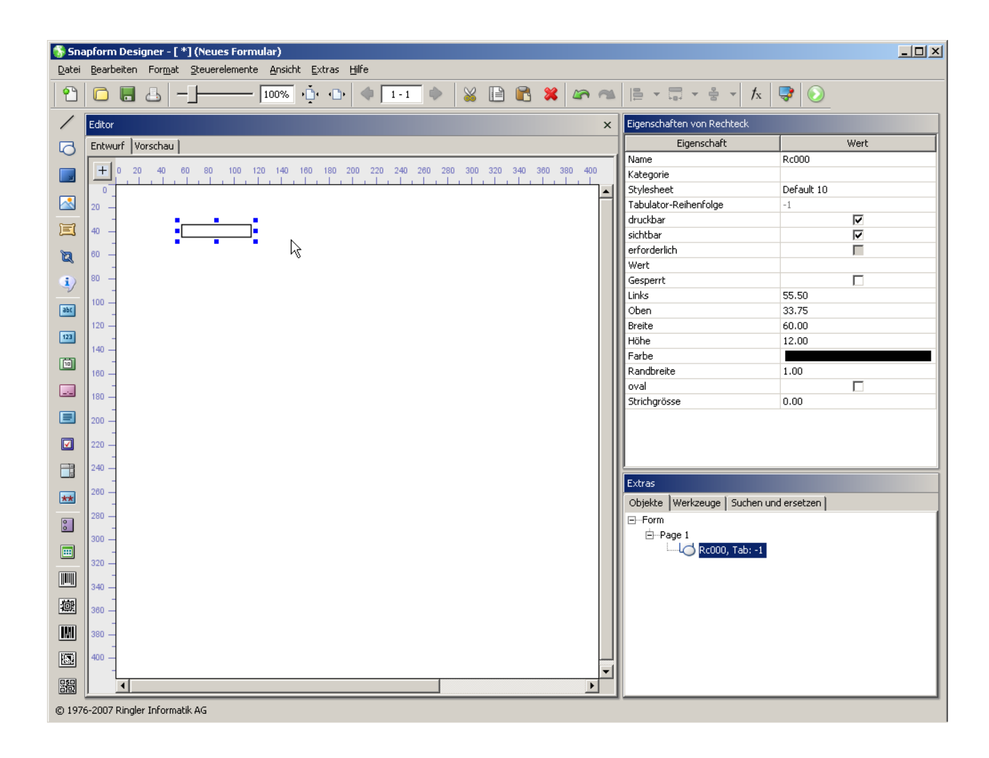
Das Rechteckwerkzeug erzeugt rechteckige und ovale Linienzüge. Nach Wahl des Werkzeugs erscheint ein Cursor mit einem Rechteck in Default-Grösse (siehe Fig. 4-26).

Fig. 4-26 *Rechteckwerkzeug-Cursor*



Mit dem Rechteckwerkzeug verknüpft ist ein schwarzes Rechteck, 60 pt breit und 12 pt hoch, mit einer Linienstärke von 1 pt. Mit Mausklick wird dieses Rechteck auf der Arbeitsfläche platziert. Dabei erhält es automatisch einen Namen, bestehend aus den Zeichen Rc und einer dreistelligen fortlaufenden Nummer, beginnend mit 000. Das erste in einem Formular mit dem Rechteckwerzeug platzierte Element trägt daher den Namen Rc000. In der Objektstruktur wird es der betreffenden Seite hinzugefügt. In Fig. 4-27 ist das erste Rechteck in einem neuen Formulars platziert, und anschliessend mit Anklicken selektiert worden.

Fig. 4-27 *Neu erstelltes Rechteck 035*



Die Länge und Breite des Rechtecks kann durch Ziehen der Ankerpunkte des Selektionsrechtecks verändert werden, die Position auch durch Anfassen und Ziehen. Beim Überfahren der Ankerpunkte ändert der Cursor in Doppelpfeile, welche die Bewegungsrichtung des Ankerpunktes angeben. Beim “Anfassen” und Drücken der Maustaste ändert sich der Cursor zu einem Hohlpfeil.

Die übrigen Merkmale eines Rechtecks werden über das Eigenschaften-Fenster kontrolliert (siehe Fig. 4-28).

Fig. 4-28 *Eigenschaften eines Rechtecks 035*

Eigenschaften von Rechteck

Eigenschaft	Wert
Name	Rc000
Kategorie	
Stylesheet	Default 10
Tabulator-Reihenfolge	-1
druckbar	☑
sichtbar	☑
erforderlich	☐
Wert	
Gesperrt	☐
Links	55.50
Oben	33.75
Breite	60.00
Höhe	12.00
Farbe	
Randbreite	1.00
oval	☐
Strichgrösse	0.00

Name

Der Name des Rechtecks im Dokument. Dieser Name kann geändert werden.

Kategorie

Zugehörigkeit zu einer oder mehreren Kategorien, welche es erlauben, logisch zusammengehörende Gruppen von Elementen zu bilden. Beim Anklicken dieses Felds öffnet sich der Kategorien Dialog (siehe Fig. 4-23).

Die Kategorien werden durch Anklicken des betreffenden Kontrollkästchens ausgewählt. Mit **OK** wird die Auswahl übernommen.

Stylesheet
Das für das Element gültige Stylesheet. Wenn explizite kein Stylesheet ausgewählt worden ist, wird das erste Default-Stylesheet genutzt. Für Rechtecke hat die Stylesheet-Definition allerdings keine Bedeutung.

Tabulator-Reihenfolge
Ordnungsnummer des Elements in der Tabulator-Reihenfolge. Dieser Wert ist immer -1 und kann nicht verändert werden. Das bedeutet, dass das Element nicht Bestandteil der Tabulatoren-Reihenfolge ist, was an sich logisch ist, da ein Rechteck nicht ein interaktives Element ist.

druckbar
Wenn das Kontrollkästchen gewählt ist, wird das Rechteck ausgedruckt.

sichtbar
Wenn das Kontrollkästchen gewählt ist, wird das Rechteck auf dem Bildschirm angezeigt.

erforderlich
Diese Eigenschaft ist bei Rechteck-Elementen immer ausgeschaltet, und kann nicht geändert werden.

Wert
Mit Anklicken dieser Eigenschaft öffnet sich der Formeleditor zur Eingabe einer Formel. Für Rechteck-Elemente wird eine Formel jedoch nur sehr selten benutzt.

Links
X-Koordinate der linken oberen Ecke der Umrandungsbox des Elements in Points. Diese Eigenschaft erlaubt eine genaue numerische Positionierung des Rechtecks.

Oben
Y-Koordinate der linken oberen Ecke der Umrandungsbox des Elements in Points. Diese Eigenschaft erlaubt eine genaue numerische Positionierung des Rechtecks.

Breite
Breite der Umrandungsbox des Rechteck-Elements.

Höhe
Höhe der Umrandungsbox des Rechteck-Elements.

Hinweis: *Die Abmessungen des Rechtecks sind Aussenmasse; sollen mehrere Rechtecke zu einer Tabelle zusammengefügt werden, muss die Linienbreite (Strichgrösse) mit berücksichtigt werden.*

Farbe

Farbe des Rechteck-Elements. Mit Anklicken der Eigenschaft öffnet sich das Farbauswahlfenster (siehe Abschnitt 3.2.9.2), welches die Definition der gewünschten Farbe erlaubt.

Linienbreite

Breite der Umrandung des Rechteck-Elements in Points.

Die Linienbreite wird vom Umrandungs-Rechteck nach innen gemessen.

Praxistip: *Rechtecke, welche schmaler sind als die Linienbreite werde unter Umständen nicht korrekt dargestellt.*

oval

Wenn dieses Kontrollkästchen angeklickt ist, wird anstelle eines Rechtecks ein Oval, welches durch das Umrandungsrechteck des Elements begrenzt ist, angezeigt.

Strichgrösse

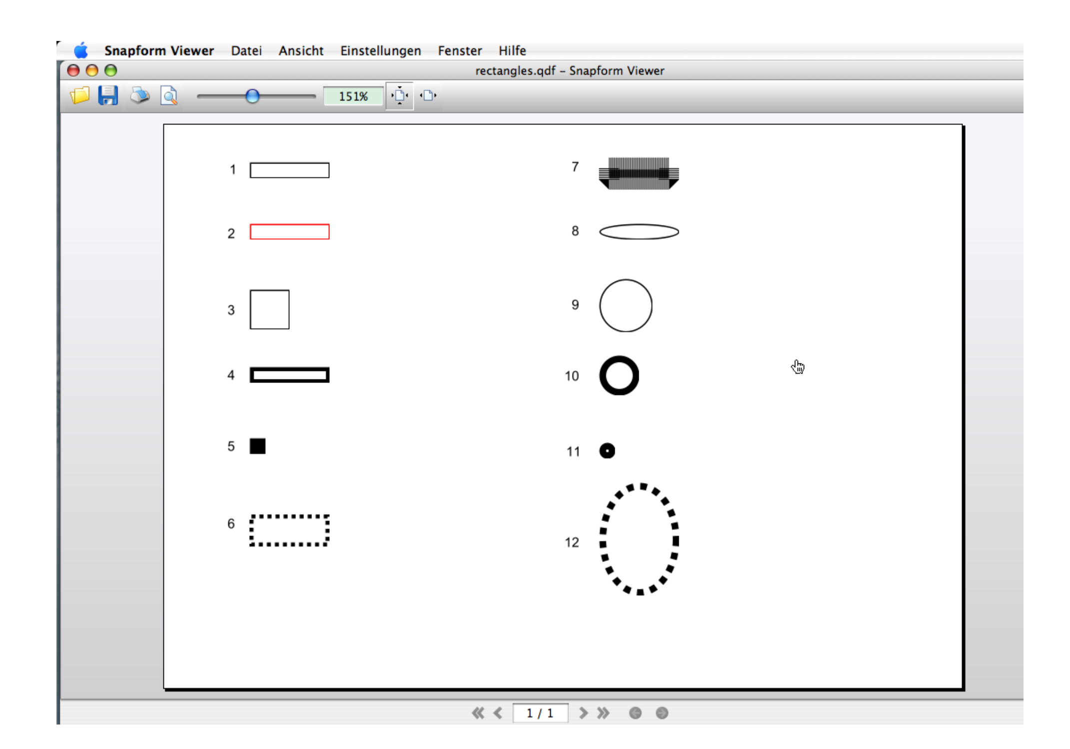
Dieser Wert bestimmt, ob die Linie gestrichelt oder ganz dargestellt ist. Der Wert 0 bedeutet, dass die Linie ganz dargestellt wird. Bei anderen Werten wird die Linie gestrichelt dargestellt, wobei der Wert die Strichlänge ausmacht.

Bei Rechtecken beginnt die Strichelung der Linie in der linken oberen Ecke des Rechtecks und läuft im Uhrzeugersinn. Bei Ovalen beginnt die Strichelung am Berührungspunkt mit der rechten Seite des Umrandungsrechtecks (3h-Position) und verläuft im Uhrzeigersinn.

Im Dokument rectangles.qdf (siehe Fig. 4-29) sind verschiedene Beispiele von Rechteck-Elementen gezeigt. Nach dem Selektieren der Rechteck-Elemente gibt das Eigenschaften-Fenster Auskunft über dessen jeweiligen Merkmale.

Hinweis: *Das Verhalten bei gestrichelten Linien lässt sich am besten im Snapform Designer mit dieser Beispielsdatei verifizieren, indem die Elemente 6 bzw. 12 gewählt und deren Grösse verändert wird.*

Fig. 4-29 *Rechtecks-Beispiele von rectangles.qdf*



1 Rc000: Default Rechteck, 60 pt breit, 12 pt hoch, Randbreite 1 pt, schwarz
2 Rc001: Rechteck, 60 pt breit, 12 pt hoch, Randbreite 1 pt, rot
3 Rc002: Quadrat (Rechteck), 40 pt breit, 40 pt hoch, Randbreite 1 pt, schwarz
4 Rc003: Rechteck, 60 pt breit, 12 pt hoch, Randbreite 3 pt, schwarz
5 Rc004: Quadrat (Rechteck), 12 pt breit, 12 pt hoch, Randbreite 6 pt, schwarz
6 Rc005: Rechteck, 60 pt breit, 24 pt hoch, Randbreite 3 pt, schwarz, Strichgrösse 3 pt
7 Rc007: Rechteck, 60 pt breit, 24 pt hoch, Randbreite 15 pt, schwarz, Strichgrösse 0.5 pt
8 Rc008: Oval, entstanden aus Default-Rechteck, Umrandungsrechteck 60 pt breit, 12 pt. hoch, Randbreite 1 pt, schwarz.
9 Rc009: Kreis (Oval), Umrandungsrechteck 40 pt breit, 40 pt hoch, Randbreite 1 pt, schwarz
10 Rc010: Kreis (Oval), Umrandungsrechteck 30 pt breit, 30 pt hoch,

Randbreite 5 pt, schwarz

11 Rc011: Kreis (Oval), Umrandungsrechteck 12 pt breit, 12 pt hoch, Randbreite 7 pt, schwarz

12 Rc012: Oval, Umrandungsrechteck 60 pt breit, 84,75 pt hoch, Randbreite 5 pt, Strichgrösse 5 pt

Praxistip: *Ein Rechteck-Element ist ein Linienzug. Eine gefülltes Rechteck kann in einem gewissen Rahmen simuliert werden, indem das Rechteck-Element einem gleich grossen Flächen-Element (siehe Abschnitt 4.3.5.1) überlagert wird.*

4.3.5.3 Flächen

Flächen-Elemente werden auch "Deckblatt" genannt, da sie im Wesentlichen zum Abdecken von Bereichen eines Hintergrundbildes dienen. Sie liegen immer zuunterst in der Element-Schichtung, und können daher keine anderen Elemente abdecken.

Das Flächenwerkzeug erzeugt rechteckige und ovale Abdeckflächen. Nach Wahl des Werkzeugs erscheint ein Cursor mit einer Abdeckfläche in Default-Grösse (siehe Fig. 4-30).

Fig. 4-30 *Flächenwerkzeug-Cursor*

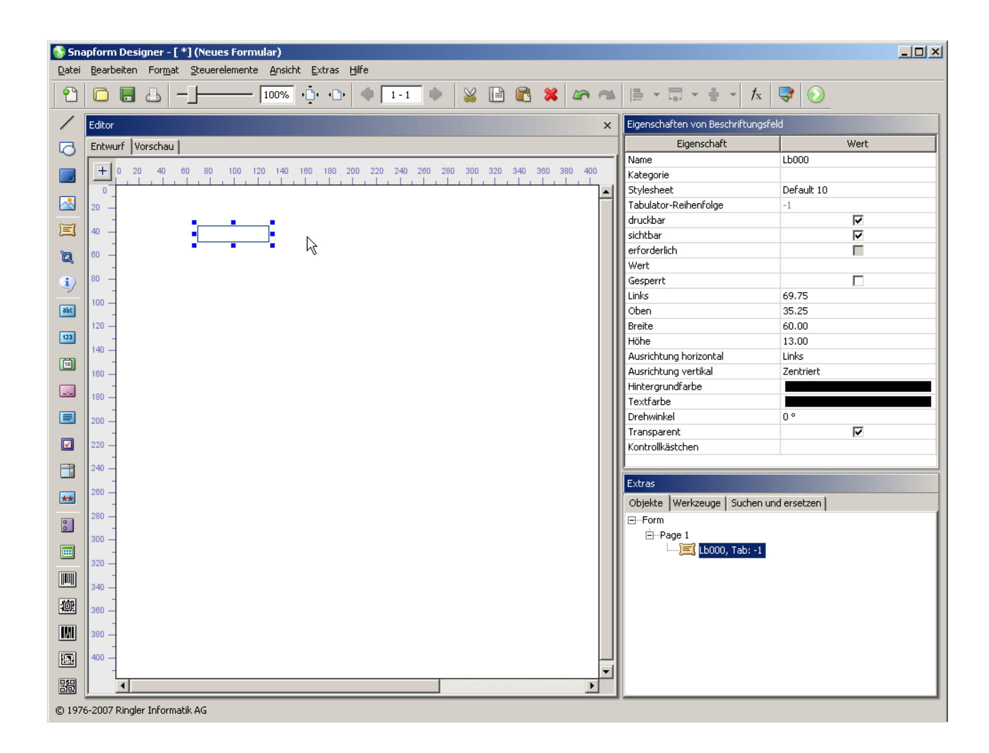


Hinweis: *Die Abdeckfläche erscheint schwarz, damit sie besser positioniert werden kann.*

Mit dem Flächenwerkzeug verknüpft ist ein schwarzes Rechteck, 60 pt breit und 12 pt hoch. Mit Mausklick wird dieses Rechteck auf der Arbeitsfläche platziert. Dabei erhält es automatisch einen Namen, bestehend aus den Zeichen Cr und einer dreistelligen fortlaufenden Nummer, beginnend mit 000. Das erste in einem Formular mit dem Flächenwerkzeug platzierte Element trägt daher den Namen Cr000. In der Objektstruktur wird es der betreffenden Seite hinzugefügt. In Fig. 4-

31 ist die erste Abdeckfläche in einem neuen Formulars platziert, und anschliessend mit Anklicken selektiert worden.

Fig. 4-31 *Neu erstellte Abdeckfläche 036*



Die Länge und Breite des Abdeck-Rechtecks kann durch Ziehen der Ankerpunkte des Selektionsrechtecks verändert werden, die Position auch durch Anfassen und Ziehen. Beim Überfahren der Ankerpunkte ändert der Cursor in Doppelpfeile, welche die Bewegungsrichtung des Ankerpunktes angeben. Beim "Anfassen" und Drücken der Maustaste ändert sich der Cursor zu einem Hohlpfeil.

Hinweis: *Bei übereinander liegenden Abdeckflächen kann es sein, dass die später platzierte Fläche als unterhalb der früheren Fläche liegend angezeigt wird. Die korrekte Schichtung der Elemente des Formulars muss in der Vorschau kontrolliert werden.*

Die übrigen Merkmale einer Abdeckfläche werden über das Eigenschaften-Fenster kontrolliert (siehe Fig. 4-32).

Fig. 4-32 *Eigenschaften einer Abdeckfläche 036*

Eigenschaften von Beschriftungsfeld

Eigenschaft	Wert
Name	Lb000
Kategorie	
Stylesheet	Default 10
Tabulator-Reihenfolge	-1
druckbar	☑
sichtbar	☑
erforderlich	☐
Wert	
Gesperrt	☐
Links	69.75
Oben	35.25
Breite	60.00
Höhe	13.00
Ausrichtung horizontal	Links
Ausrichtung vertikal	Zentriert
Hintergrundfarbe	
Textfarbe	
Drehwinkel	0 °
Transparent	☑
Kontrollkästchen	

Name

Der Name der Abdeckfläche im Dokument. Dieser Name kann geändert werden.

Kategorie

Zugehörigkeit zu einer oder mehreren Kategorien, welche es erlauben, logisch zusammengehörende Gruppen von Elementen zu bilden. Beim Anklicken dieses Felds öffnet sich der Kategorien Dialog (siehe Fig. 4-23).

Die Kategorien werden durch Anklicken des betreffenden Kontrollkästchens ausgewählt. Mit **OK** wird die Auswahl übernommen.

Stylesheet

Das für das Element gültige Stylesheet. Wenn explizite kein Stylesheet ausgewählt worden ist, wird das erste Default-Stylesheet genutzt. Für

Abdeckflächen hat die Stylesheet-Definition allerdings keine Bedeutung.

Tabulator-Reihenfolge

Ordnungsnummer des Elements in der Tabulator-Reihenfolge. Dieser Wert ist immer -1 und kann nicht verändert werden. Das bedeutet, dass das Element nicht Bestandteil der Tabulatoren-Reihenfolge ist, was an sich logisch ist, da eine Abdeckfläche nicht ein interaktives Element ist.

druckbar

Wenn das Kontrollkästchen gewählt ist, wird die Abdeckfläche ausgedruckt.

sichtbar

Wenn das Kontrollkästchen gewählt ist, wird die Abdeckfläche auf dem Bildschirm angezeigt.

erforderlich

Diese Eigenschaft ist bei Abdeckflächen-Elementen immer ausgeschaltet, und kann nicht geändert werden.

Wert

Mit Anklicken dieser Eigenschaft öffnet sich der Formeleditor zur Eingabe einer Formel. Für Abdeckflächen-Elemente wird eine Formel jedoch nur sehr selten benutzt.

Gesperrt

Wenn das Kontrollkästchen gewählt ist, wird die Abdeckfläche blockiert, und kann nicht (versehentlich) verschoben werden. Zudem werden sämtliche übrigen Eigenschaften im Eigenschaften-Dialog deaktiviert. Der Zugriff auf die Eigenschaften ist erst wieder möglich, wenn dieses Kontrollkästchen deaktiviert wird.

Links

X-Koordinate der linken oberen Ecke der Umrandungsbox des Elements in Points. Diese Eigenschaft erlaubt eine genaue numerische Positionierung der Abdeckfläche.

Oben

Y-Koordinate der linken oberen Ecke der Umrandungsbox des Elements in Points. Diese Eigenschaft erlaubt eine genaue numerische Positionierung der Abdeckfläche.

Breite

Breite der Umrandungsbox des Abdeckflächen-Elements.

Höhe

Höhe der Umrandungsbox des Abdeckflächen-Elements.

oval

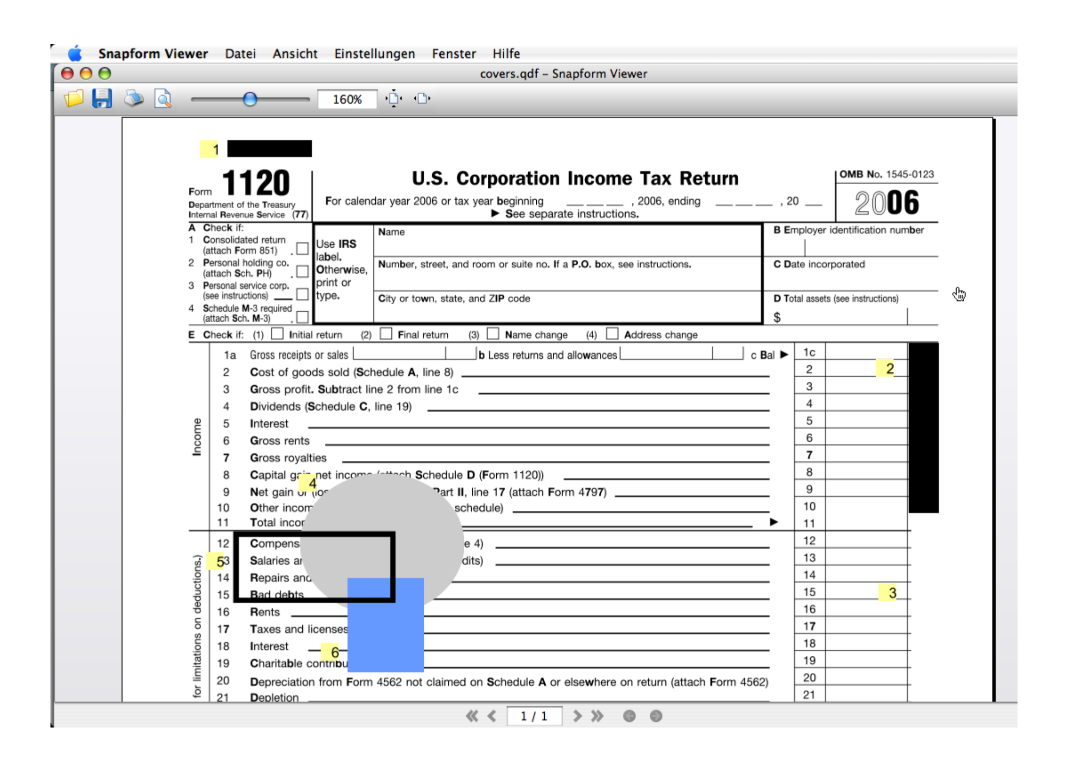
Wenn dieses Kontrollkästchen angeklickt ist, wird anstelle eines Rechtecks ein Oval, welches durch das Umrandungsrechteck des Elements begrenzt ist, angezeigt.

Hintergrundfarbe

Farbe des Abdeckflächen-Elements. Mit Anklicken der Eigenschaft öffnet sich das Farbauswahlfenster (siehe Abschnitt 3.2.9.2), welches die Definition der gewünschten Farbe erlaubt.

Im Dokument covers.qdf (siehe Fig. 4-33) sind verschiedene Beispiele von Abdeckflächen-Elementen gezeigt. Nach dem Selektieren der Abdeckflächen-Elemente gibt das Eigenschaften-Fenster Auskunft über dessen jeweiligen Merkmale.

Fig. 4-33 *Abdeckflächen-Beispiele von covers.qdf*



Damit alle Elemente sichtbar sind, zeigt Fig. 4-34 dasselbe Dokument in der Entwicklungsansicht in Snapform Designer. Das eigentliche Abdeckelement (Position 3 im Dokument) ist dabei selektiert. Zudem zeigt das Fenster **Extras –> Objekte** die Elementstruktur des Beispielformulars.

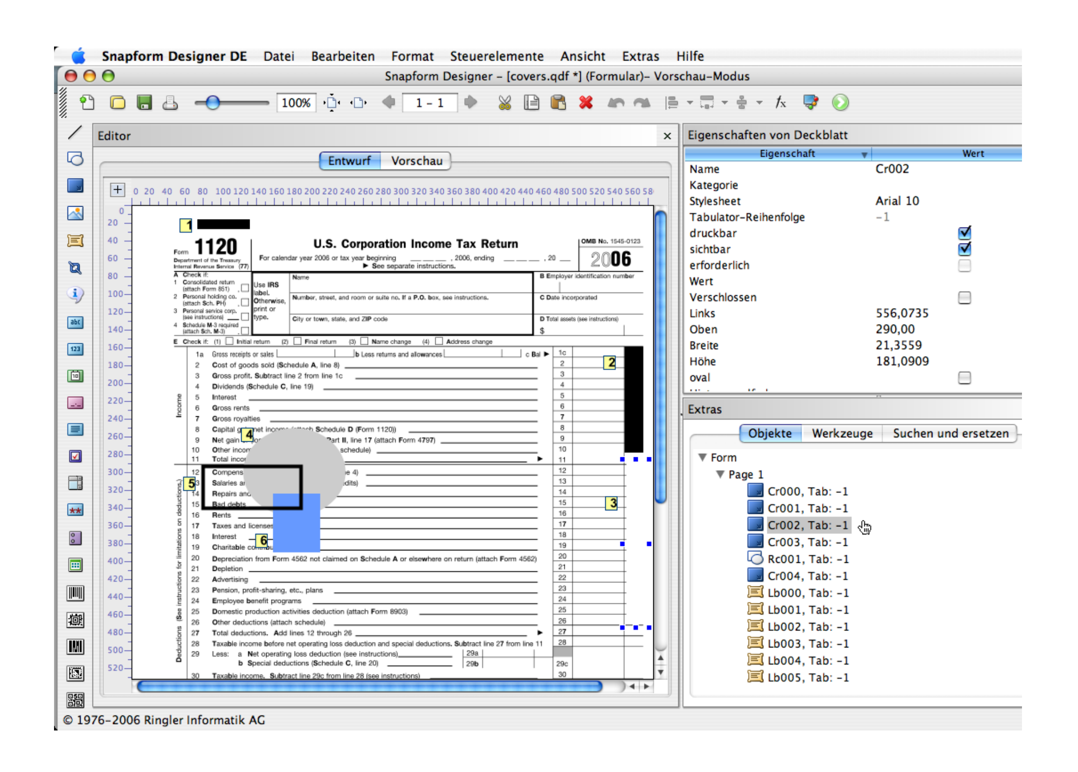


Fig. 4-34 *Abdeckflächen-Beispiele von covers.qdf in der Entwicklungsansicht*

1 Cr000: Default Abdeckfläche, 60 pt breit, 12 pt hoch, schwarz
2 Cr001: Abdeckfläche rechteckig, 21.35 pt breit, 120 pt hoch, schwarz
3 Cr002: Abdeckfläche rechteckig, 21.35 pt breit, 181.1 pt hoch, weiss
4 Cr003: Abdeckfläche oval, Umrandungsrechteck 115.9 pt breit, 99.6 pt hoch, hellgrau
5 Rc001: Rechteck, 114.9 pt breit, 51.2 pt hoch, schwarz, Linienbreite 4 pt
6 Cr004: Abdeckfläche rechteckig, 53.9 pt breit, 66.5 pt hoch, blau

Die Positionen 4, 5 und 6 (Elemente Cr003, Rc001 und Cr004) zeigen die Überlappungen. Das Element Cr003 liegt unterhalb des Elements Cr004, da letzteres später erzeugt wurde. Das Element Rc001 wurde nach dem Element Cr003, aber vor dem Element Cr004 erzeugt. Da Abdeckflächen immer "zuunterst" im Elementstapel liegen, überdeckt das Rechteck-Element beide Abdeckflächen.

4.3.5.4 Bilder und Logos

Ein wesentliches Element des Grundlayouts von Formularen sind Bilder und Logos. In Snapform können Rastergrafiken als Bilder in ein Bild-Element eingefügt werden, wobei sie entweder fest in das Dokument

hineingebunden werden können, oder dynamisch (hauptsächlich beim Öffnen des Dokuments) geladen werden.

Der Vorteil eines fix eingebundenen Bildes ist, dass dieses immer zur Verfügung steht, unabhängig davon, ob der Snapform Viewer eine Netzwerkverbindung nutzen kann, oder ob das Bild lokal verfügbar ist.

Snapform kann die meisten in einer Office- oder einer Web-Umgebung benutzten Rastergrafik-Formate nutzen. Sollte ein Format nicht unterstützt werden, gibt es eine grosse Anzahl von Konvertern. Wenn alles fehlschlägt, besteht zudem die Möglichkeit des Erstellens eines PDFs und das Extrahieren des Bildes in Acrobat (Professional).

Grundsätzlich muss bei Rasteformaten beachtet werden, dass es einen direkten Zusammenhang zwischen der Grösse eines Bildes in Bildpunkten (Pixeln), der Auflösung des Bildes (in Bildpunkten pro Zoll (dots per inch) sowie der absoluten Dimensionen des Bildes auf dem Formular (in Zoll oder Millimetern) gibt. Diese Gesetzmässigkeiten können nicht durchbrochen werden.

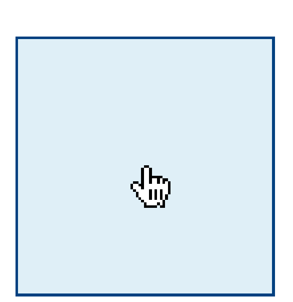
Ein Rasterbild hat immer eine bestimmte Grösse in Bildpunkten. Diese Grösse hat sich beim Erstellen des Bildes ergeben. Wird das betreffende Bild nun in ein Bild-Element eingefügt, erhält es eine bestimmte absolute Grösse. Daraus resultiert sofort die Auflösung des Bildes für den betreffenden Fall. Diese Auflösung gilt nun als Mass, in welcher Qualität das Bild angezeigt (oder ausgedruckt) wird. Bei Farb- und Graustufen-Bildern werden folgende Auflösungen als ausreichend betrachtet: 72 dpi für Bildschirmanzeigen, 150 dpi für Office-Druckern, 300 dpi für Offsetdruck. Bei Schwarz-Weiss-Bildern gelten folgende Auflösungen als ausreichend: 72 dpi für Bildschirmdarstellung, 300 dpi für Office-Drucker, 1200 dpi für Offsetdruck. Diese Auflösungen sollten andererseits nicht allzustark überschritten werden, denn das betreffende Ausgabemedium kann viel höhere Auflösungen nicht verarbeiten. Die Folge von zu hohen Auflösungen ist dann eine unnötig grosse Datei bzw. unnötig lange Datenübermittlungszeiten.

Snapform Designer unterstützt die folgenden Grafikformate bei Bild-Elementen: JPEG, BMP, TIFF, GIF und PNG.

Hinweis: *JPEG ist ein sehr kompaktes Format, benutzt aber eine verlustbehaftete Kompression, welche bei starken Kontrasten zu Artefakten und Unschärfen führt. Starke Kontraste treten besonders bei gescanntem Text und Linien auf. Für diese Art von Bildern ist das JPEG-Format ungeeignet. Anstelle dessen sollte TIFF, PNG oder GIF verwendet werden.*

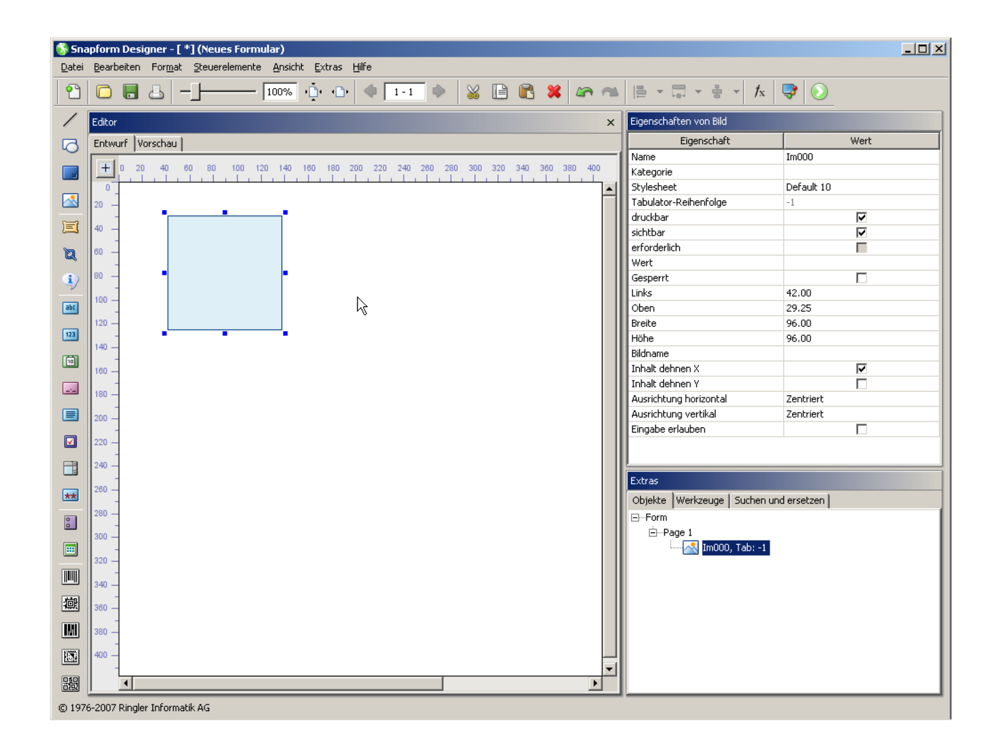
Das Bildwerkzeug erzeugt ein Feld, in welches eine Rastergrafik geladen werden kann. Nach Wahl des Werkzeugs erscheint ein Cursor mit einem Bild-Element in Default-Grösse (siehe Fig. 4-35).

Fig. 4-35 *Bildwerkzeug-Cursor*



Mit dem Bildwerkzeug verknüpft ist ein Bildfeld, 96 pt breit und 96 pt hoch. Mit Mausklick wird dieses Feld auf der Arbeitsfläche platziert. Dabei erhält es automatisch einen Namen, bestehend aus den Zeichen Im und einer dreistelligen fortlaufenden Nummer, beginnend mit 000. Das erste in einem Formular mit dem Bildwerkzeug platzierte Element trägt daher den Namen Im000. In der Objektstruktur wird es der betreffenden Seite hinzugefügt. In Fig. 4-36 ist das erste Bildfeld in einem neuen Formulars platziert, und anschliessend mit Anklicken selektiert worden.

Fig. 4-36 *Neu erstelltes Bildfeld 037*



Die Länge und Breite des Bildfelds kann durch Ziehen der Ankerpunkte des Selektionsrechtecks verändert werden, die Position auch durch Anfassen und Ziehen. Beim Überfahren der Ankerpunkte ändert der Cursor in Doppelpfeile, welche die Bewegungsrichtung des Ankerpunktes angeben. Beim "Anfassen" und Drücken der Maustaste ändert sich der Cursor zu einem Hohlpfeil.

Die übrigen Merkmale einer Bildfeldes werden über das Eigenschaften-Fenster kontrolliert (siehe Fig. 4-37).

Fig. 4-37 *Eigenschaften eines Bildfeldes 037*

Eigenschaften von Bild

Eigenschaft	Wert
Name	Im000
Kategorie	
Stylesheet	Default 10
Tabulator-Reihenfolge	-1
druckbar	☑
sichtbar	☑
erforderlich	☐
Wert	
Gesperrt	☐
Links	42.00
Oben	29.25
Breite	96.00
Höhe	96.00
Bildname	
Inhalt dehnen X	☑
Inhalt dehnen Y	☐
Ausrichtung horizontal	Zentriert
Ausrichtung vertikal	Zentriert
Eingabe erlauben	☐

Name

Der Name des Bildfeldes im Dokument. Dieser Name kann geändert werden.

Kategorie

Zugehörigkeit zu einer oder mehreren Kategorien, welche es erlauben, logisch zusammengehörende Gruppen von Elementen zu bilden. Beim Anklicken dieses Felds öffnet sich der Kategorien Dialog (siehe Fig. 4-23).

Die Kategorien werden durch Anklicken des betreffenden Kontrollkästchens ausgewählt. Mit **OK** wird die Auswahl übernommen.

Stylesheet

Das für das Element gültige Stylesheet. Wenn explizite kein Stylesheet ausgewählt worden ist, wird das erste Default-Stylesheet genutzt. Bei

Bildfeldern wird von den Stylesheet-Definitionen einzig Randeinstellungen in Allgemeine Einstellungen berücksichtigt.

Tabulator-Reihenfolge

Ordnungsnummer des Elements in der Tabulator-Reihenfolge. Dieser Wert wird automatisch vergeben, und liegt in der Reihe der Feld-Tabulator-Reihenfolge (welche mit 10 beginnend in Zehnerschritten erhöht wird. Das erste Feld des Formulars hat daher den Tabulator-Reihenfolge-Wert 10, das zweite den Wert 20, etc. Bei Bildfeldern kann der Wert der Tabulator-Reihenfolge nicht verändert werden, da Bildfelder nicht in die Tabulator-Reihenfolge eingebunden sind.

druckbar

Wenn das Kontrollkästchen gewählt ist, erscheint das Bildfeld im Ausdruck.

sichtbar

Wenn das Kontrollkästchen gewählt ist, erscheint das Bildfeld auf dem Bildschirm.

erforderlich

Diese Eigenschaft ist bei Bildfeldern immer ausgeschaltet, und kann nicht geändert werden.

Wert

Mit Anklicken dieser Eigenschaft öffnet sich der Formeleditor zur Eingabe einer Formel.

In Bildfeldern mit referenziertem Bildinhalt (das heisst, das Bild wird nicht direkt in das Feld eingefügt) ist das Resultat der Formel der Pfad zur Bilddatei. Der Pfad ist ein URL (Uniform Resource Locator), welcher im Fall von Bildern hauptsächlich eines der folgenden Protokolle umfassen kann: *http*, *https*, *ftp*, *file*. Damit die ersten drei Protokolle benutzt werden können, muss beim Einsatz des Formulars eine Internet-Verbindung bestehen oder aufgebaut werden können. Das Protokoll file bezieht sich auf eine lokale Datei, was bedeutet, dass der Pfad dieser Datei entweder berechnet werden kann, oder konstant sein muss.

Das einfachste Beispiel ist die Definition eines konstanten Wertes (Position 2 im Musterdokument images.qdf, siehe Fig. 4-39):

```
http://www.google.ch/images/logo_sm.gif
```

Dies ist ein konstanter Wert, und beim Öffnen des Formulars wird das definierte Bild vom Internet geladen.

Sehr ähnlich ist (im Musterdokument nicht abgebildet) ein konstantes Resultat:

```
="http://www.google.ch/images/logo_sm.gif"
```

Dies lädt beim Öffnen des Formulars sowie bei jeder generellen Neuberechnung das betreffende Bild.

Es ist ebenso möglich, indirekt zu referenzieren. In diesem Fall wird der URL in einem Textfeld gehalten, und dessen Wert übernommen (Position 3 im Musterdokument images.qdf, siehe Fig. 4-39). Die Formel hierzu lautet:

```
=Text000
```

Wobei der URL sich im Textfeld Tx000 befinden muss.

Als Resultat einer etwas komplexeren Berechnung ist die Referenzierung über ein Kontrollkästchen (Position 5 im Musterdokument images.qdf, siehe Fig. 4-39). Die Formel hierzu lautet:

```
=if(Cb000, "http://www.snapform.com/img/s-blue.jpg",
"http://www.snapform.com/img/s-green.jpg")
```

Wenn das Kontrollkästchen Cb000 angeklickt ist, wird die Datei s-blue.jpg geladen (der blaue Gecko), sonst wird die Datei s-green.jpg geladen (der grüne Gecko).

Eine noch ausgeklügeltere Logik ist in Position 6 des Musterdokuments images.qdf gezeigt.

Gesperrt

Wenn das Kontrollkästchen gewählt ist, wird das Bildfeld im Snapform Designer blockiert, und kann nicht (versehentlich) verschoben werden. Zudem werden sämtliche übrigen Eigenschaften im Eigenschaften-Dialog deaktiviert. Der Zugriff auf die Eigenschaften ist erst wieder möglich, wenn dieses Kontrollkästchen deaktiviert wird.

Links

X-Koordinate der linken oberen Ecke der Umrandungsbox des Elements in Points. Diese Eigenschaft erlaubt eine genaue numerische Positionierung des Bildfeldes.

Oben

Y-Koordinate der linken oberen Ecke der Umrandungsbox des Elements in Points. Diese Eigenschaft erlaubt eine genaue numerische Positionierung des Bildfeldes.

Breite

Breite des Bildfeldes in Points.

Höhe

Höhe des Bildfeldes in Points.

Bildname

Wenn das Bild direkt in das Feld eingebunden werden soll, wird in diesem Eigenschaftsfeld dessen Name eingegeben. Wenn das Feld angeklickt wird, erscheint ein Drop Down Menu mit den Namen der bereits im Formular enthaltenen Bilder. Mit dem Anklicken der Schaltfläche **...** öffnet sich ein normaler Datei Öffnen-Dialog, welcher es erlaubt, das betreffende Bild auszuwählen.

Ein über diese Option eingefügtes Bild wird im Dokument gespeichert, und kann für weitere Bildfelder wieder verwendet werden. Eine übliche Anwendung dieser Eigenschaft ist bei Logos.

Inhalt dehnen X und Inhalt dehnen Y

In den seltensten Fällen ist das eingefügte Bild genau so gross wie das Bildfeld. Mit diesen Kontrollkästchen wird gesteuert, ob das Bild in Originalgrösse dargestellt werden soll, oder ob es soweit skaliert werden soll, dass es das Bildfeld in in einer oder beiden Richtungen ausfüllt. Dabei ergeben sich folgende Kombinationen:

- Beide Kontrollkästchen aus:

  Das Bild wird in Originalgrösse dargestellt. Dabei werden nicht in das Bildfeld passende Bereiche abgeschnitten.

- **Inhalt dehnen X** ein und **Inhalt dehnen Y** aus:

Das Bild wird proportional skaliert, und zwar so, dass es ganz in das Bildfeld passt. Wenn es proportional breiter als das Bildfeld ist, entspricht die Breite derjenigen des Bildfelds.

- **Inhalt dehnen X** aus und **Inhalt dehnen Y** ein:

  Das Bild wird proportional skaliert, und zwar so, dass es ganz in das Bildfeld passt. Wenn es proportional höher als das Bildfeld ist, entspricht die Höhe derjenigen des Bildfelds.

- Beide Kontrollkästchen ein:

  Das Bild wird in beiden Richtungen so skaliert, dass es das Bildfeld vollständig ausfüllt. Dabei werden die Proportionen des Bildes verändert.

Ausrichtung horizontal

Wenn das Bild schmaler als das Bildfeld dargestellt wird, bestimmt diese Eigenschaft, wie es im Bildfeld positioniert wird.

Wenn der Eigenschaftswert angeklickt wird, erscheint ein Drop Down Menu mit der Auswahl **links**, **zentriert**, **rechts**, welche das Bild links-, mitten- bzw. rechtszentriert im Bildfeld positioniert. Grundeinstellung ist **zentriert**.

Ausrichtung vertikal

Wenn das Bild wenigher hoch als das Bildfeld dargestellt wird, bestimmt diese Eigenschaft, wie es im Bildfeld positioniert wird.

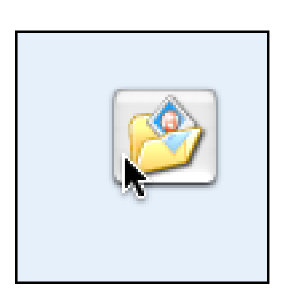
Wenn der Eigenschaftswert angeklickt wird, erscheint ein Drop Down Menu mit der Auswahl **oben**, **zentriert**, **unten**, welche das Bild oben-, mitten- bzw. unten-zentriert im Bildfeld positioniert. Grundeinstellung ist **zentriert**.

Eingabe erlauben

Wenn dieses Kontrollkästchen gewählt ist, kann der Benutzer sein eigenes Bild in das Feld einfügen. Eine typische Anwendung ist beispielsweise das Hinzufügen eines Passbildes bei einem Personalformular. Im Musterdokument images.qdf ist Position 4 ein

derartiges Eingabefeld. Wenn der Cursor über ein derartiges Feld kommt, erscheint ein Bildeingabe-Cursor(siehe Fig. 4-38).

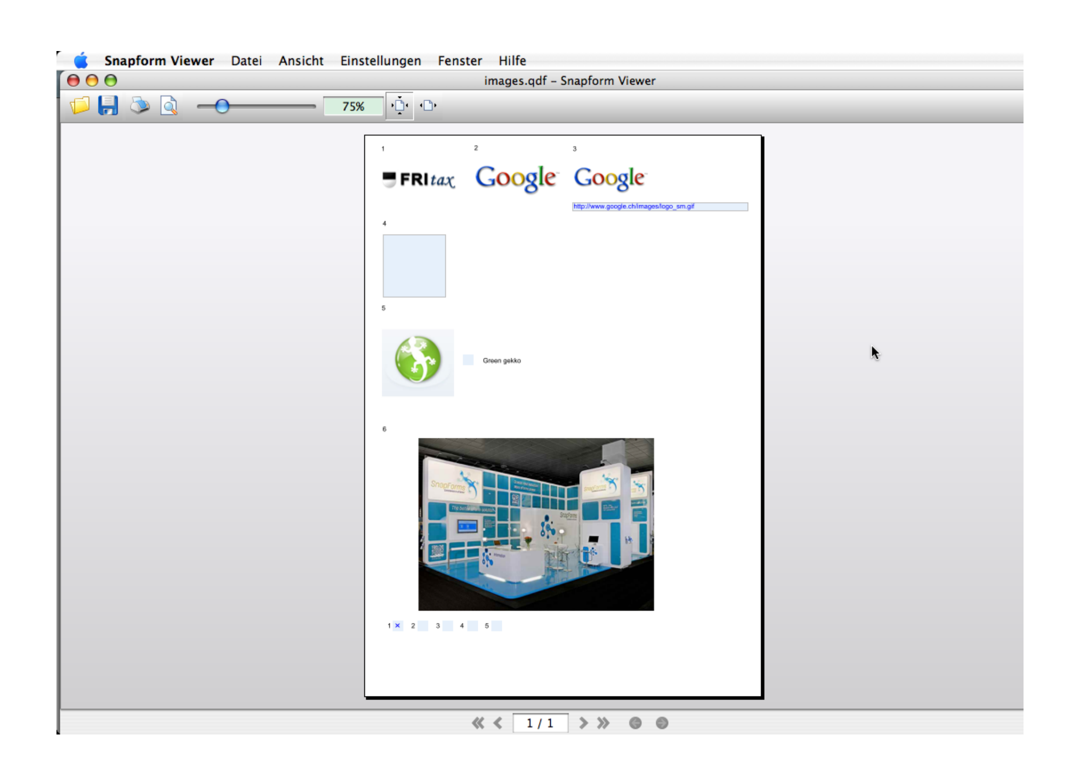
Fig. 4-38 *Bildeingabe-Cursor*



Wenn der Benutzer nun dieses Feld anklickt, öffnet sich ein Datei Öffnen-Dialog, mit welchem das einzufügende Bild ausgewählt werden kann. Dieses Bild wird fest in das Formular eingebunden, und bleibt nach dem Speichern erhalten.

Im Dokument images.qdf (siehe Fig. 4-39) sind verschiedene Beispiele von Bildfeldern gezeigt. Nach dem Selektieren der Bildfelder gibt das Eigenschaften-Fenster bzw. der Formeleditor Auskunft über dessen jeweiligen Merkmale.

Fig. 4-39 *Bildfelder-Beispiele von images.qdf*



1 Im000: Bildfeld, 113,5 pt breit, 60,7 pt hoch, Bild start_panel_top.png fest eingebunden
2 Im001: Bildfeld, 129,3 pt breit, 63,7 pt hoch, URL von Bild als Konstante definiert
3 Im003: Bildfeld, 113,6 pt breit, 60,7 pt hoch, URL von Bild in Textfeld Tx000 definiert; indirekte Referenz
4 Im005: Bildfeld, 96 pt breit, 96 pt hoch, Benutzereingabe erlaubt
5 Im002: Bildfeld, 109,7 pt breit, 133,2 pt hoch, URL über Kontrollkästchen Cb000 gesteuert
6 Im004: Bildfeld, 465,3 pt breit, 260 pt hoch, URL über Satz von 5 Optionsfeldern gesteuert.

Die Formeln für die Positionen 2, 3 und 5 sind weiter oben aufgeführt, und können auch durch Doppelklick auf die betreffenden Felder in der Entwurfsansicht von Snapform Designer im Formeleditor angezeigt werden.

4.3.6 Text

Test ist ein wesentlicher Bestandteil von Formularen. Text gehört zum Teil zum Grundlayout, zum Teil zur Formularebene. In diesem Abschnitt wird zur Grundebene gehörender Text behandelt.

Zur Anzeige von Text im Grundlayout wird das Beschriftungsfeld-Element (auch Label genannt) benutzt. Beschriftungsfeld-Elemente können unformatierten oder formatierten Text enthalten.

Hinweis: *Beschriftungsfelder hatten in früheren Snapform-Versionen neben der Darstellung von Text eine weitere Funktion. Sie dienten als Container für Formeln. Diese Anwendung gilt als überholt, und sollte nicht mehr weiter genutzt werden.*

Unformatierter Text besteht aus einer einzigen Schriftart und Schriftgrösse, mit einem einzigen Attribut. Diese Art Text kann als einfacher Text eingegeben werden, oder aber als HTML. Die HTML-Form ist für gewisse Merkmale, wie vertikale Zentrierung im Feld oder für gewisse Textattribute, welche nicht über die Element-Eigenschaften eingestellt werden können, notwendig.

Formatierter Text kann aus mehreren Schriftgrössen und -arten bestehen, und verschiedene Attribute umfassen. Formatierter Text wird in HTML-Form eingegeben. Es werden alle HTML-Tags unterstützt, welche nicht auf externe Elemente abstützen. Ebensowenig werden externe Stylesheets unterstützt (siehe auch Abschnitt 4.3.6.2).

Hinweis: *Bei der Verwendung von Text in HTML-Form ist zu beachten, dass der Darstellungsmechanismus von HTML-Text eine geringere Auflösung als derjenige für einfachen Text erzeugt. Es kann daher sein, dass HTML-Text im Ausdruck nicht dem gewohnten Qualitätsniveau entspricht.*

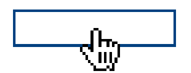
Praxistip: *Die Eingabe von Text in HTML-Form in den Formeleditor kann zum Teil etwas mühsam sein. Es wird daher empfohlen, einen externen Texteditor zu benutzen, welcher die HTML-Formatierung unterstützt.*

*Eine weitere Vereinfachung der Eingabe bieten Utilities wie Markdown (***Werkzeugs***), welche eine vereinfachte Syntax zur Erstellung von Texten benutzen, und diese dann im Texteditor in HTML-Code umwandeln.*

4.3.6.1 Das Beschriftungsfeld-Werkzeug

Das Beschriftungsfeld-Werkzeug erzeugt ein Feld, für die Anzeige von Text. Nach Wahl des Werkzeugs erscheint ein Cursor mit Textrahmen in Default-Grösse (siehe Fig. 4-40).

Fig. 4-40 *Beschriftungsfeld-Werkzeug-Cursor*



Mit dem Beschriftungsfeld-Werkzeug verknüpft ist ein Rechteck, 60 pt breit und 13 pt hoch. Mit Mausklick wird dieses Rechteck auf der Arbeitsfläche platziert. Dabei erhält es automatisch einen Namen, bestehend aus den Zeichen Lb und einer dreistelligen fortlaufenden Nummer, beginnend mit 000. Das erste in einem Formular mit dem Beschriftungsfeld-Werkzeug platzierte Element trägt daher den Namen Lb000. In der Objektstruktur wird es der betreffenden Seite hinzugefügt.

In Fig. 4-41 ist das erste Beschriftungsfeld in einem neuen Formulars platziert, und anschliessend mit Anklicken selektiert worden.

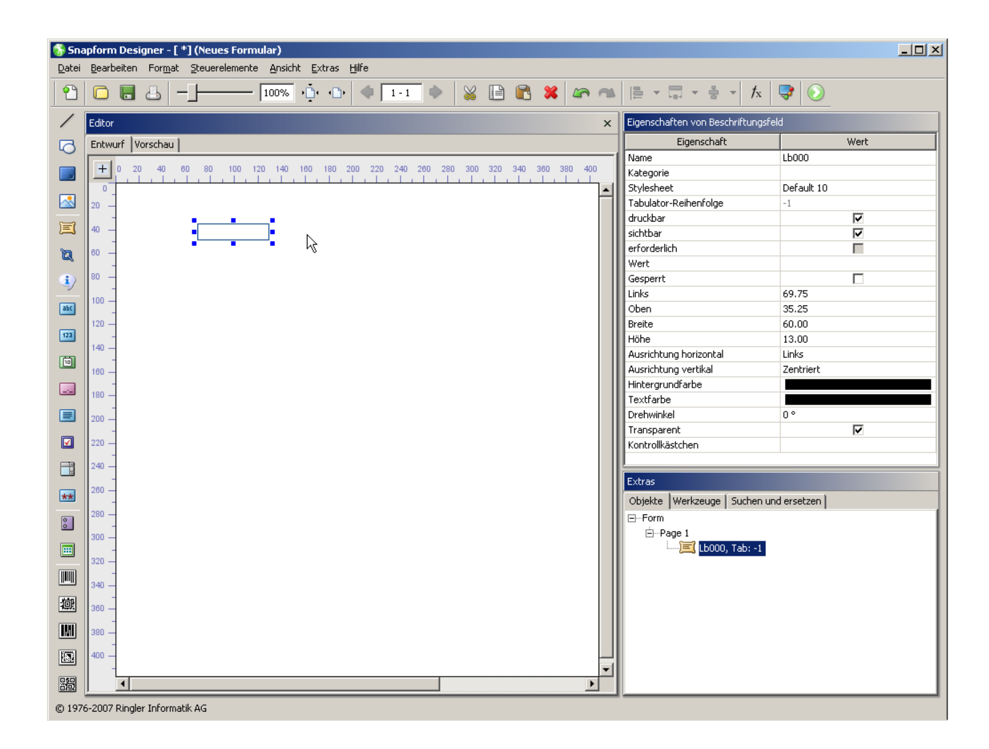


Fig. 4-41 *Neu erstelltes Beschriftungsfeld 038*

Die Länge und Breite Beschriftungsfelds kann durch Ziehen der Ankerpunkte des Selektionsrechtecks verändert werden, die Position auch durch Anfassen und Ziehen. Beim Überfahren der Ankerpunkte ändert der Cursor in Doppelpfeile, welche die Bewegungsrichtung des Ankerpunktes angeben. Beim “Anfassen” und Drücken der Maustaste ändert sich der Cursor zu einem Hohlpfeil.

Die übrigen Merkmale eines Beschriftungsfelds werden über das Eigenschaften-Fenster kontrolliert (siehe Fig. 4-42).

Fig. 4-42 *Eigenschaften eines Beschriftungsfelds 038*

Eigenschaften von Beschriftungsfeld

Eigenschaft	Wert
Name	Lb000
Kategorie	
Stylesheet	Default 10
Tabulator-Reihenfolge	-1
druckbar	☑
sichtbar	☑
erforderlich	☐
Wert	
Gesperrt	☐
Links	69.75
Oben	35.25
Breite	60.00
Höhe	13.00
Ausrichtung horizontal	Links
Ausrichtung vertikal	Zentriert
Hintergrundfarbe	
Textfarbe	
Drehwinkel	0 °
Transparent	☑
Kontrollkästchen	

Name

Der Name des Beschriftungsfelds im Dokument. Dieser Name kann geändert werden.

Kategorie

Zugehörigkeit zu einer oder mehreren Kategorien, welche es erlauben, logisch zusammengehörende Gruppen von Elementen zu bilden. Beim Anklicken dieses Felds öffnet sich der Kategorien Dialog (siehe Fig. 4-23).

Die Kategorien werden durch Anklicken des betreffenden Kontrollkästchens ausgewählt. Mit **OK** wird die Auswahl übernommen.

Stylesheet

Das für das Element gültige Stylesheet. Wenn explizite kein Stylesheet ausgewählt worden ist, wird das erste Default-Stylesheet genutzt. Von

den Stylesheet-Definitionen sind für Beschriftungsfelder die Bereiche **Schriftarteinstellungen** und **Erweiterte Sprachunterstützung** in **Allgemeine Einstellungen** des Stylesheet-Editors wichtig.

Hinweis: *Diese Stylesheet-Definitionen stehen in keinem Zusammenhang mit Cascading Stylesheets in HTML, und haben keinen Einfluss auf die Darstellung von Text in HTML-Form.*

Tabulator-Reihenfolge

Ordnungsnummer des Elements in der Tabulator-Reihenfolge. Dieser Wert ist immer -1 und kann nicht verändert werden. Das bedeutet, dass das Element nicht Bestandteil der Tabulatoren-Reihenfolge ist, was an sich logisch ist, da ein Beschriftungsfeld nicht ein interaktives Element ist.

druckbar

Wenn das Kontrollkästchen gewählt ist, wird das Beschriftungsfeld ausgedruckt.

sichtbar

Wenn das Kontrollkästchen gewählt ist, wird das Beschriftungsfeld auf dem Bildschirm angezeigt.

erforderlich

Diese Eigenschaft ist bei Beschriftungsfeldern immer ausgeschaltet, und kann nicht geändert werden.

Wert

In dieser Eigenschaft des Beschriftungsfeldes wird der anzuzeigende Text eingegeben. Mit Anklicken dieser Eigenschaft öffnet sich der Formeleditor zur Texteingabe.

Unformatierter Text wird direkt, ohne Anführungszeichen oder Gleichheitszeichen eingegeben. HTML-Code kann ebenso direkt eingegeben werden. Falls der anzuzeigende Text das Resultat einer Berechnung ist (zum Beispiel die Kombination von konstantem Text mit einem Feldwert), muss ein Gleichheitszeichen eingegeben werden, und der Text (einschliesslich allfälliger HTML-Code) muss zwischen Anführungszeichen gesetzt werden.

HTML-Code wird, sofern es sich nicht um Berechnungen oder komplexen Code handelt, auch im Entwurfs-Modus direkt interpretiert, und angezeigt. Wenn der HTML-Code komplex ist (Tabellen, Bilder) oder das Resultat einer Berechnung ist, kann der Text nicht direkt angezeigt werden. Zur Anzeige dieses Texts muss in dien Vorschau-Modus umgeschaltet werden. In diesem Fall wird im Entwurfsmodus der Text der Formel angezeigt.

Weitere Beispiele von Beschriftungsfeldern mit Text in HTML befinden sich im Abschnitt 4.3.6.2.

Gesperrt

Wenn das Kontrollkästchen gewählt ist, wird das Beschriftungsfeld im Snapform Designer blockiert, und kann nicht (versehentlich) verschoben werden. Zudem werden sämtliche übrigen Eigenschaften im Eigenschaften-Dialog deaktiviert. Der Zugriff auf die Eigenschaften ist erst wieder möglich, wenn dieses Kontrollkästchen deaktiviert wird.

Links

X-Koordinate der linken oberen Ecke der Umrandungsbox des Elements in Points. Diese Eigenschaft erlaubt eine genaue numerische Positionierung des Beschriftungsfelds.

Oben

Y-Koordinate der linken oberen Ecke der Umrandungsbox des Elements in Points. Diese Eigenschaft erlaubt eine genaue numerische Positionierung des Beschriftungsfelds.

Breite

Breite der Umrandungsbox des Beschriftungsfelds.

Höhe

Höhe der Umrandungsbox des Beschriftungsfelds.

Ausrichtung horizontal

Diese Eigenschaft bestimmt, wie der Text im Beschriftungsfeld ausgerichtet wird.

Wenn der Eigenschaftswert angeklickt wird, erscheint ein Drop Down Menu mit der Auswahl **links**, **zentriert**, **rechts**, welche den Text

linksbündig, mittenzentriert bzw. rechtsbündig im Beschriftungsfeld positioniert. Grundeinstellung ist **links**.

Bei mehrzeiligen unformatierten Texten wird jede einzelne Zeile entsprechend ausgerichtet. In der HTML-Form wird die längste Zeile eines Textblocks gemäss der Eigenschaft ausgerichtet. Die übrigen Zeilen werden dann linksbündig auf diese Zeile ausgerichtet.

Ausrichtung vertikal

Diese Eigenschaft bestimmt, wie der Text im Beschriftungsfeld vertikal positioniert wird.

Wenn der Eigenschaftswert angeklickt wird, erscheint ein Drop Down Menu mit der Auswahl **oben**, **zentriert**, **unten**, welche den Text am oberen Feldrand, mittenzentriert bzw. am unteren Feldrand ausrichtet. Grundeinstellung ist **zentriert**.

Diese Eigenschaft wird nur bei Texten in HTML-Form vollständig berücksichtigt. In diesem Fall wird der HTML-Textblock entsprechend ausgerichtet.

1-zeilige unformatierte Texte werden immer am unteren Feldrand ausgerichtet. Mehrzeilige unformatierte Texte werden immer am oberen Feldrand ausgerichtet.

Hintergrundfarbe

Farbe des Beschriftungsfeldes. Mit Anklicken der Eigenschaft öffnet sich das Farbauswahlfenster (siehe Abschnitt 3.2.9.2), welches die Definition der gewünschten Farbe erlaubt.

Die Hintergrundfarbe wird nur dann berücksichtigt, wenn das Kontrollkästchen **Transparent** ausgeschaltet ist.

Textfarbe

Farbe des Texts. Mit Anklicken der Eigenschaft öffnet sich das Farbauswahlfenster (siehe Abschnitt 3.2.9.2), welches die Definition der gewünschten Farbe erlaubt.

Drehwinkel

Der Text im Beschriftungsfeld kann um Vielfache von 90° gedreht werden. Beim Anklicken dieser Eigenschaft öffnet sich ein Drop Down Menu mit der Auswahl der Drehwinkel 0°, 90°, 180°, 270°. Grundeinstellung ist 0°.

Transparent

Der Hintergrund des Beschriftungsfeldes wird mit diesem Kontrollkästchen auf transparent (keine deckende Hintergrundsfarbe) geschaltet. Diese Eigenschaft ist in der Grundeinstellung eingeschaltet.

Kontrollkästchen

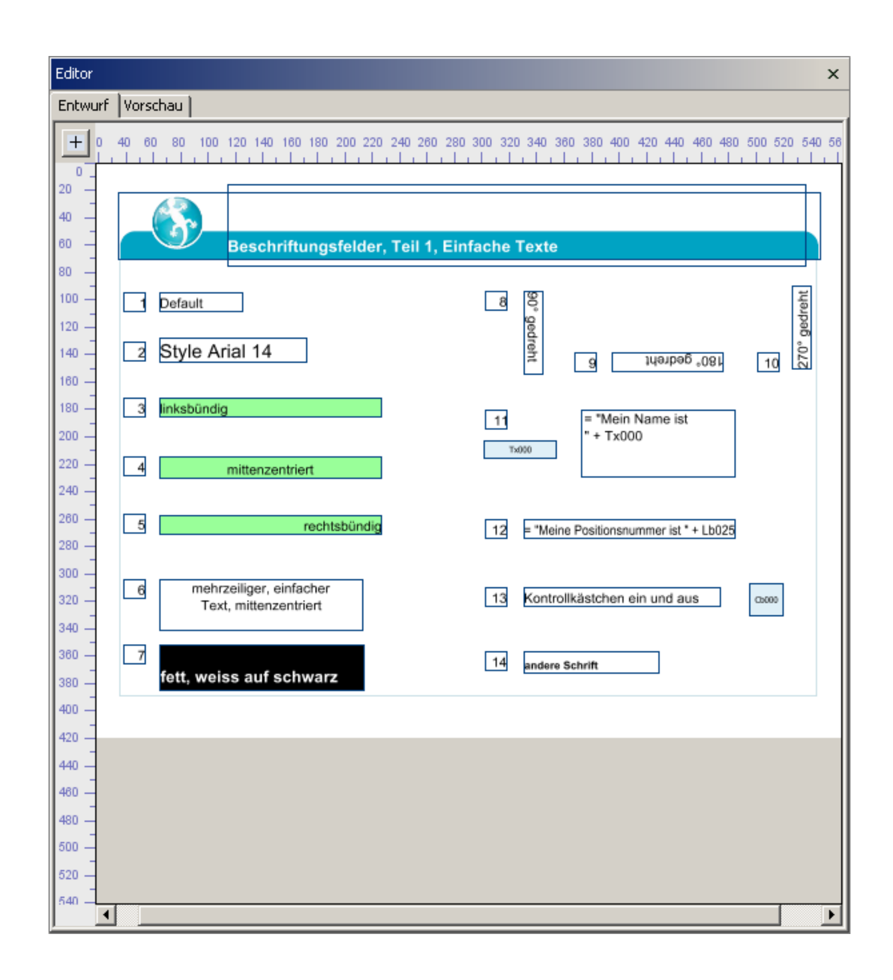
Es ist möglich, das Beschriftungsfeld mit einem Kontrollkästchen zu assoziieren. Das Beschriftungsfeld wird dadurch anklickbar und zu einer "Erweiterung" des Kontrollkästchens. Wenn das Beschriftungsfeld angeklickt wird, ändert sich der Zustand des Kontrollkästchens.

Beim Anklicken dieser Eigenschaft öffnet sich ein Drop Down Menu mit der Auswahl sämtlicher Kontrollkästchen des Dokuments. Die Wahl des betreffenden Feldnamens setzt die Assoziation fest.

Im Dokument labels1.qdf (siehe Fig. 4-43 (Entwurfsmodus)) bzw. Fig. 4-44 (Ansicht in Snapform Viewer)) sind verschiedene Beispiele von Beschriftungsfeldern mit einfachem Text gezeigt. Das Dokument labels2.qdf (siehe Fig. 4-45 (Entwurfsmodus) bzw. Fig. 4-46 (Ansicht in Snapform Viewer)) zeigt Beispiele von Beschriftungsfeldern mit formatiertem (HTML) Text.

Nach dem Selektieren der Beschriftungsfelder im Snapform Designer gibt das Eigenschaften-Fenster Auskunft über dessen jeweiligen Merkmale. Ein Doppelklick auf das Feld öffnet den Formeleditor zur Anzeige des Texts.

Fig. 4-43 *Beschriftungsfelder-Beispiele von labels1.qdf, im Entwurfs-Modus 039*



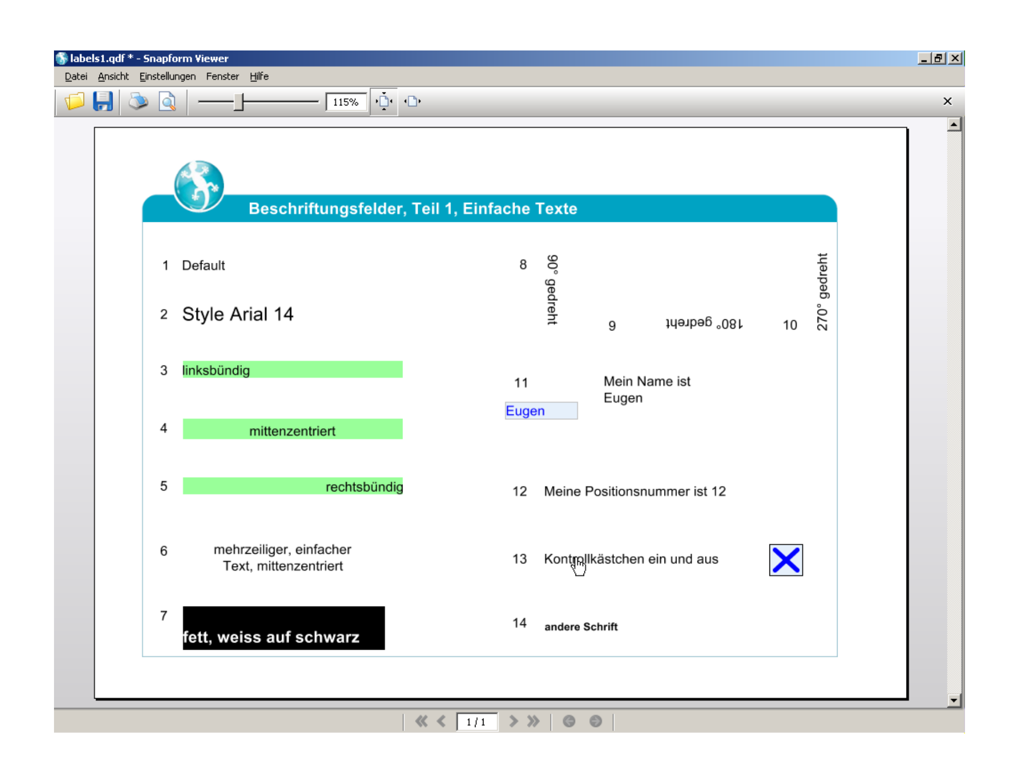


Fig. 4-44 *Beschriftungsfelder-Beispiele von labels1.qdf in Snapform Viewer 040*

1 Lb000: Beschriftungsfeld, Grundeinstellung

2 Lb001: Beschriftungsfeld, Schriftgrösse über Stylesheet eingestellt

3 Lb002: Beschriftungsfeld, Text linksbündig, Hintergrundfarbe grün (zur besseren Visualisierung der Feldgrösse), nicht transparent

4 Lb003: Beschriftungsfeld, Text mittenzentriert, Hintergrundfarbe grün (zur besseren Visualisierung der Feldgrösse), nicht transparent

5 Lb004: Beschriftungsfeld, Text rechtsbündig, Hintergrundfarbe grün (zur besseren Visualisierung der Feldgrösse), nicht transparent

6 Lb005: Beschriftungsfeld, mehrzeiliger Text, wird auf den oberen Rand des Feldes ausgerichtet

7 Lb006: Beschriftungsfeld, Schrift fett über Stylesheet eingestellt, einzeiliger Text wird auf den unteren Rand des Feldes ausgerichtet

8 Lb007: Beschriftungsfeld, Text um 90° gedreht

9 Lb008: Beschriftungsfeld, Text um 180° gedreht

10 Lb009: Beschriftungsfeld, Text um 270° gedreht

11 Lb010: Beschriftungsfeld, Text ist aus konstantem Teil und Feldwert von Feld Tx000 zusammengestellt; der Wert von Feld Tx000 wird beim Verlassen in das Beschriftungsfeld übertragen

12 Lb011: Beschriftungsfeld, Text ist aus konstantem Teil und Wert des Beschriftungsfelds Lb025 zusammengestellt.

13 Lb012: Beschriftungsfeld, mit Kontrollkästchen Cb000 assoziiert; das Beschriftungsfeld wird aktiv, und Anklicken ändert den Status des

Kontrollkästchens

14 Lb013: Beschriftungsfeld, Schriftart über Stylesheet eingestellt

Fig. 4-45 *Beschriftungsfelder-Beispiele von labels2.qdf, im Entwurfs-Modus 041*

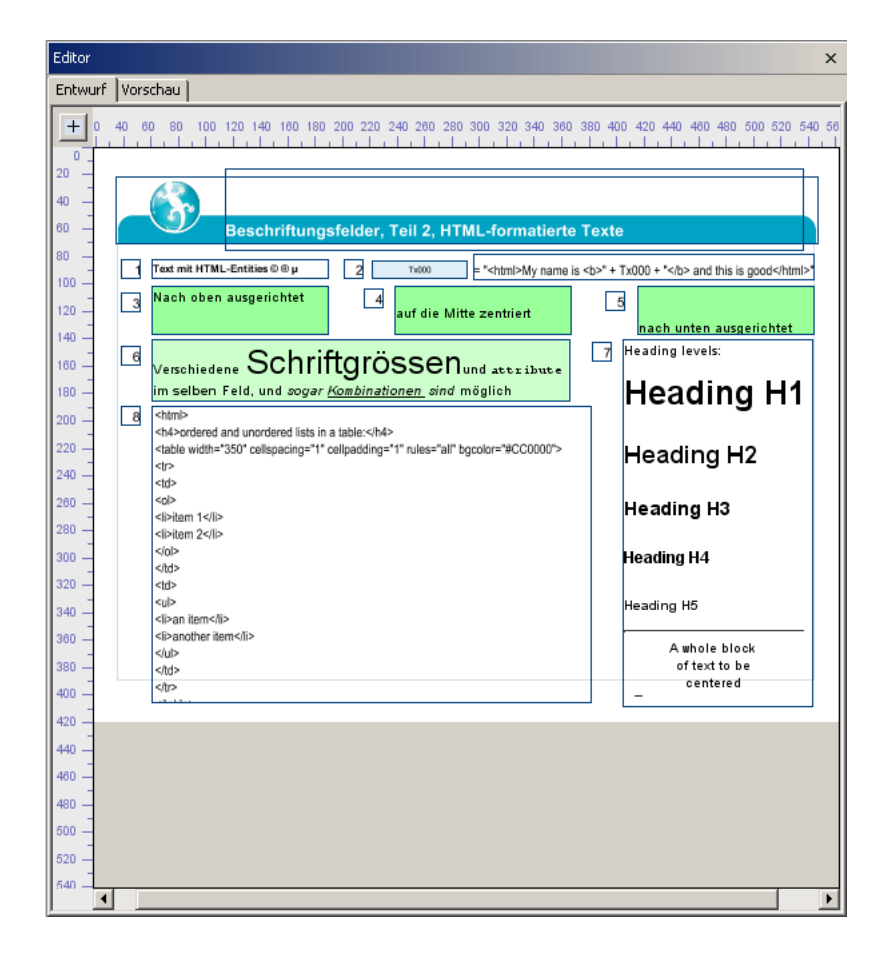
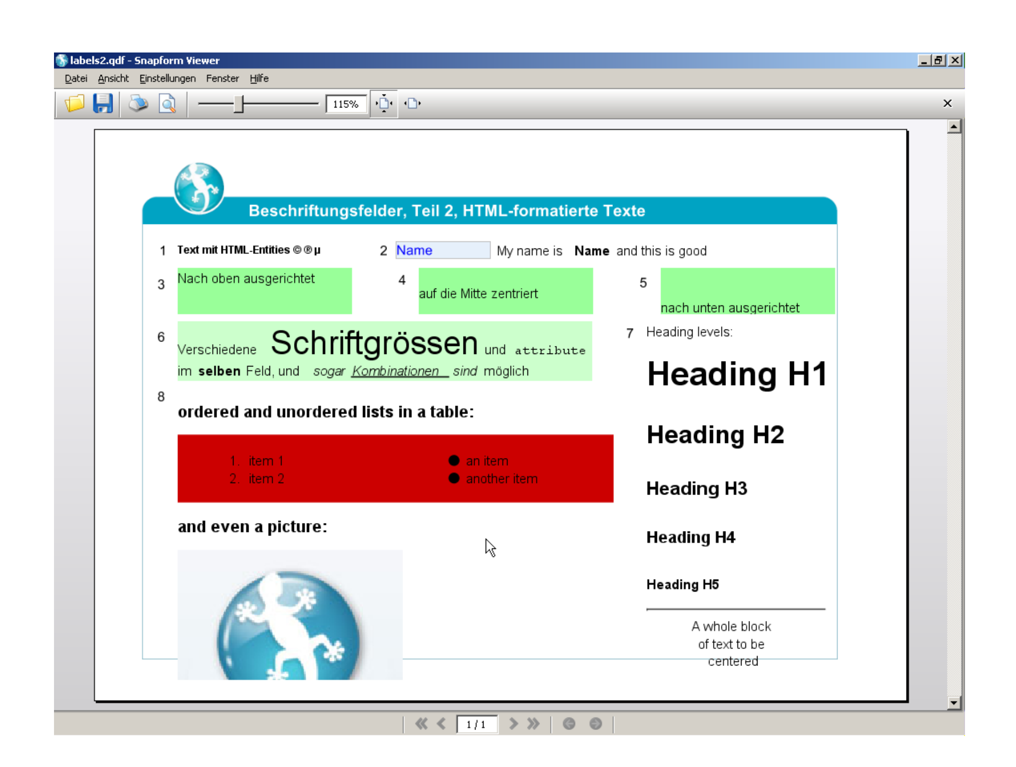


Fig. 4-46 *Beschriftungsfelder-Beispiele von labels2.qdf in Snapform Viewer 042*



1 Lb000: Beschriftungsfeld, Text in HTML, mit Sonderzeichen als HTML-Entities

2 Lb007: Beschriftungsfeld, Text ist aus konstantem Teil und Feldwert von Feld Tx000 zusammengestellt; der Wert von Feld Tx000 wird beim Verlassen in das Beschriftungsfeld übertragen und fett angezeigt

3 Lb001: Beschriftungsfeld, Text in HTML, auf den oberen Feldrand ausgerichtet, Hintergrundfarbe grün (zur besseren Visualisierung der Feldgrösse), nicht transparent

4 Lb002: Beschriftungsfeld, Text in HTML, auf die Feldmitte ausgerichtet, Hintergrundfarbe grün (zur besseren Anzeige der Feldgrösse), nicht transparent

5 Lb003: Beschriftungsfeld, Text in HTML, auf den unteren Feldrand ausgerichtet, Hintergrundfarbe grün (zur besseren Visualisierung der Feldgrösse), nicht transparent

6 Lb004: Beschriftungsfeld, Text in HTML, mit verschiedenen wechselnden inline-Textattributen versehen

7 Lb005: Beschriftungsfeld, Text in HTML, mit Heading-Levels, horizontaler Trennlinie und zentriertem Text

8 Lb006: Beschriftungsfeld, Text in HTML, mit Listen, Tabellen und eingebettetem Bild

4.3.6.2 In Snapform verwendbare HTML-Tags und Attribute

Snapform unterstützt weitgehend die Tags und Attribute, welche im HTML 3.2-Standard definiert sind. Neuere Konstrukte werden nicht unterstützt. Zum Teil wird in einem solchen Fall der HTML-Code wörtlich angezeigt, zum Teil überhaupt nicht.

Die offizielle HTML-Spezifikation (in englischer Sprache) befindet sich unter http://www.w3c.org/TR/REC-html32 .

4.4 Formularebene

Über dem Grundlayout aufgebaut ist die Formularebene. Die Formularebene umfasst aktive Elemente. Das bedeutet, dass diese Elemente auf Benutzeraktionen reagieren (zum Beispiel, dass Werte in ein Feld eingegeben werden, oder dass ein Kontrollkästchen angeklickt wird).

Ganz vereinfacht gesagt, zur Formularebene gehören Elemente, deren Tab-Reihenfolge einen Wert grösser als 0 aufweist, und geändert werden kann. Dies umfasst auch das Bildfeld-Element (siehe Abschnitt 4.3.5.4). Da hier die Interaktivität jedoch nicht allzuhäufig genutzt wird, ist dieses Element dem Grundlayout zugeordnet.

Andererseits werden Beschriftungsfelder, welche normalerweise dem Grundlayout zugeordnet werden, in ihrer Funktion als Formelcontainer auch als Bestandteil der Formularebene betrachtet.

Die Elemente der Formularebene können für sich allein gespeichert werden. Das Dateiformat benutzt die Endung .hdo. Um die Formularebene zu speichern, wird im normalen Speichern unter Dialog die Option **Snapform Feldebene (.hdo)** gewählt. Eine gespeicherte Formularebene wird mit dem Menubefehl **Datei –> Feldebene öffnen** geladen.

4.4.1 Labels und Formelcontainer

Beschriftungsfelder (Labels) und Formelcontainer sind dasselbe, je nach ihrer Anwendung. In früheren Snapform-Versionen waren einzig Beschriftungsfelder für die Aufnahme von Formeln eingerichtet. In der aktuellen Version können Formeln jedem Element des Formulars zugewiesen werden, und Funktionen gelten für das ganze Dokument. Aus diesen Gründen ist die Bedeutung der Beschriftungsfelder als Formelcontainer nur noch von historischer Natur.

Hinweis: *Es wird empfohlen, Formulare mit Beschriftungsfeldern als Formelcontainer zu überarbeiten, und Funktionen zu benutzen (mehr zu Funktionen in Abschnitt 4.5.1.2).*

4.4.2 Eingabefelder

Eine der Hauptfunktionen eines Formulars ist die Datenerfassung. Die Datenerfassung erfolgt normalerweise über Eingabefelder. Snapform benutzt Textfelder im weitesten Sinn, Optionsfelder und Kontrollkästchen sowie Auswahllisten zur Datenerfassung. Textfelder sind weiter in spezialisierte Felder (z.B. Daten oder Eingabemasken) aufgeteilt.

In den nachstehenden Abschnitten werden erst die Gemeinsamkeiten von Textfeldern beschrieben, anschliessend die spezifischen Eigenheiten der spezialisierten Felder.

4.4.2.1 Eingabefelder im Allgemeinen

Eingabefelder werden nach Wahl des jeweiligen Werkzeugs auf der Arbeitsfläche platziert. Sie werden der Dokumentstruktur hinzugefügt, und sie werden in die Tab-Reihenfolge eingefügt. Die in Fig. 4-47 gezeigten und nachstehend beschriebenen Eigenschaften sind für alle Eingabefelder gleich (ausser einzelne Merkmale, welche in Passwortfeldern nicht vorkommen).

Fig. 4-47 *Eigenschaften von Eingabefeldern am Beispiel eines Textfeldes 043*

Eigenschaften von Textfeld

Eigenschaft	Wert
Name	Tx000
Kategorie	
Stylesheet	Default 10
Tabulator-Reihenfolge	10
druckbar	☑
sichtbar	☑
erforderlich	☐
Wert	
Gesperrt	☐
Links	118.8312
Oben	75.00
Breite	96.00
Höhe	12.00
schreibgeschützt	☐
Einmalfeld	☐
Kurzinfo-Text	
Ausrichtung horizontal	Links
Länge	0
Eingabefelder konfigurieren	...
Link zu Objekt	
Nur Zahlen	☐

Name

Der Name des Eingabefelds im Dokument. Dieser Name kann geändert werden.

Kategorie

Zugehörigkeit zu einer oder mehreren Kategorien, welche es erlauben, logisch zusammengehörende Gruppen von Elementen zu bilden. Beim Anklicken dieses Felds öffnet sich der Kategorien Dialog (siehe Fig. 4-23).

Die Kategorien werden durch Anklicken des betreffenden Kontrollkästchens ausgewählt. Mit **OK** wird die Auswahl übernommen.

Stylesheet

Das für das Element gültige Stylesheet. Wenn explizite kein Stylesheet ausgewählt worden ist, wird das erste Default-Stylesheet genutzt. Für Eingabefelder sind die folgenden Stylesheet-Definitionen von Belang: **Allgemeine Einstellungen** und **Eingabefelder**. Die Stylesheets sind zur Hauptsache für die Darstellung von verschiedenen Zuständen von Eingabefeldern gemacht worden.

Tabulator-Reihenfolge

Ordnungsnummer des Elements in der Tabulator-Reihenfolge. Dieser Wert wird automatisch vergeben, und liegt in der Reihe der Feld-Tabulator-Reihenfolge (welche mit 10 beginnend in Zehnerschritten erhöht wird). Das erste Feld des Formulars hat daher den Tabulator-Reihenfolge-Wert 10, das zweite den Wert 20, etc.. Der Wert kann verändert werden, so dass eine frei bestimmbare Tabulatorreihenfolge aufgebaut werden kann, welche nicht mit der Reihenfolge des Hinzufügens der Felder übereinstimmen muss.

druckbar

Wenn das Kontrollkästchen gewählt ist, wird das Eingabefeld ausgedruckt.

sichtbar

Wenn das Kontrollkästchen gewählt ist, wird das Eingabefeld auf dem Bildschirm angezeigt.

erforderlich

Wenn das Kontrollkästchen gewählt ist, muss in diesem Eingabefeld zwingend eine Eingabe erfolgt sein, damit das Formular gültig ist (für Formular-Gültigkeit, siehe auch Abschnitt 4.8.4.2).

Wert

Diese Eigenschaft kann bei Eingabefeldern für zwei verschiedene Zwecke benutzt werden. Einerseits wird damit der Grundwert des Feldes festgelegt (indem ein einfacher Wert eingegeben wird), andererseits kann auch eine Formel definiert werden, deren Resultat im Feld angezeigt wird.

Ein einfach eingegebener Wert ist ein Grundwert, welcher beim Öffnen eines leeren Formulars angezeigt und beim eigentlichen Ausfüllen

überschrieben wird. Ein einfacher Wert hat als Formel das folgende Aussehen:

```
Dies ist der Grundwert
```

Beim Rücksetzen des Formulars (Schaltfläche Eingabewerte löschen in Snapform Viewer) wird Grundwert wieder angezeigt.

Ein als Formel eingegebener Grundwert

```
= "Grundwert als Formel"
```

wird beim Öffnen des Dokuments so angezeigt. Beim Überschreiben beim eigentlichen Ausfüllen des Formulars wird das Feld jedoch als "übersteuertes Resultat" markiert. Beim Rücksetzen wird der Feldinhalt geleert, wobei das Feld nach wie vor als "übersteuert" markiert bleibt. Erst mit Anklicken der Feld-Rücksetzmarke wird der Grundwert angezeigt.

Hinweis: *Eingabefelder unterstützten HTML-formatierten Text nicht.*

Eine Formel wird beim Öffnen des Dokuments sowie bei Formular-Aktionen ausgeführt. Wenn das Feld nicht schreibgeschützt ist, kann der Feldwert überschrieben werden. Das Feld wird jedoch als "übersteuert" markiert.

Gesperrt

Wenn das Kontrollkästchen gewählt ist, wird das Eingabefeld im Snapform Designer blockiert, und kann nicht (versehentlich) verschoben werden. Zudem werden sämtliche übrigen Eigenschaften im Eigenschaften-Dialog deaktiviert. Der Zugriff auf die Eigenschaften ist erst wieder möglich, wenn dieses Kontrollkästchen deaktiviert wird.

Links

X-Koordinate der linken oberen Ecke der Umrandungsbox des Elements in Points. Diese Eigenschaft erlaubt eine genaue numerische Positionierung des Eingabefelds.

Oben

Y-Koordinate der linken oberen Ecke der Umrandungsbox des Elements in Points. Diese Eigenschaft erlaubt eine genaue numerische Positionierung des Eingabefelds.

Breite

Breite der Umrandungsbox des Eingabefelds.

Höhe

Höhe der Umrandungsbox des Eingabefelds.

schreibgeschützt

Wenn das Kontrollkästchen gewählt ist, ist das Eingabefeld als schreibgeschützt markiert, und dessen Inhalt kann nicht manuell überschrieben werden. Der Wert kann nach wie vor über Formeln verändert werden. Ein schreibgeschütztes Eingabefeld ist nach wie vor in der Tab-Reihenfolge eingebunden, und der Wert kann mit **<Ctrl><C>** bzw. **<Cmd><C>** in die Zwischenablage kopiert werden.

Hinweis: *Bei Passwortfeldern ist diese Eigenschaft nicht verfügbar.*

Einmalfeld

Es gibt Informationen, welche nach dem ursprünglichen Ausfüllen des Formulars nicht mehr verändert werden dürfen und daher geschützt werden müssen. In Snapform erfolgt dieser Schutz mit Einmalfeldern, welche beim Ausfüllen bis zum ersten Speichern verändert werden können, anschliessend aber als schreibgeschützt markiert werden.

Wenn dieses Kontrollkästchen gewählt ist, kann das Eingabefeld bis zum ersten Speichern eines ausgefüllten Formulars verändert werden. Nach dem ersten Speichern des Formulars wird das Eingabefeld als schreibgeschützt markiert.

Hinweis: *Bei Passwortfeldern ist diese Eigenschaft nicht verfügbar.*

Kurzinfo-Text

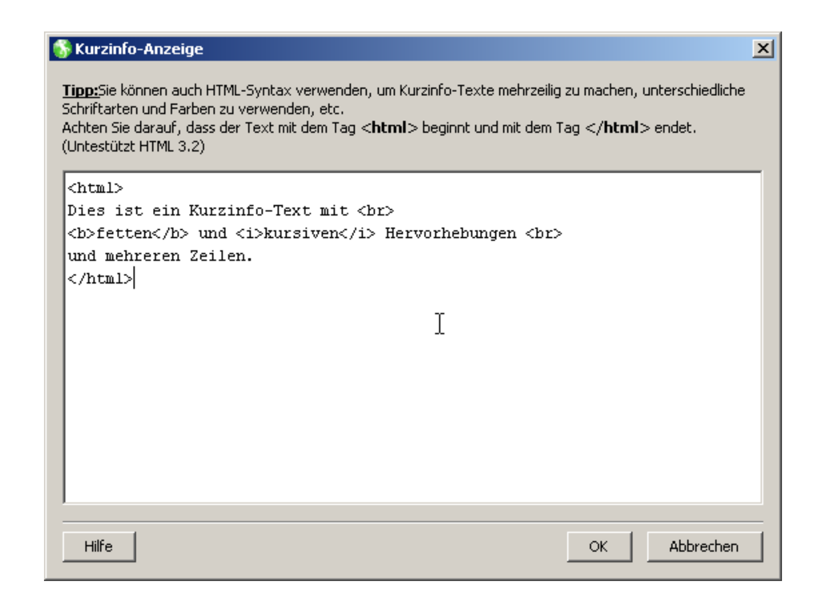
Die Kurzinfo ist ein Hilfetext, welcher erscheint, wenn der Mauscursor während einer gewissen Zeit über dem Feld liegt. Dieser Text ist in vielen Fällen die erste und einfachste Hilfe zum betreffenden Feld. Mit zunehmenden Anforderungen an die Barrierefreiheit von elektronischen Formularen erhält der Kurzinfo-Text eine zunehmend

grössere Bedeutung, da Screenreader diese Information benutzen. Es ist daher, besonders für Formulare im Behörden-Umfeld, wichtig, einen gut abgefassten Kurzinfo-Text einzufügen.

Wenn diese Eigenschaft gewählt wird, öffnet sich das Kurzinfo-Anzeigefenster (siehe Fig. 4-48). Wie der Erläuterungstext angibt, können für die Kurzinfos auch HTML-Texte eingegeben werden. Es gelten hierfür die gleichen Bedingungen wie für Beschreibungsfelder.

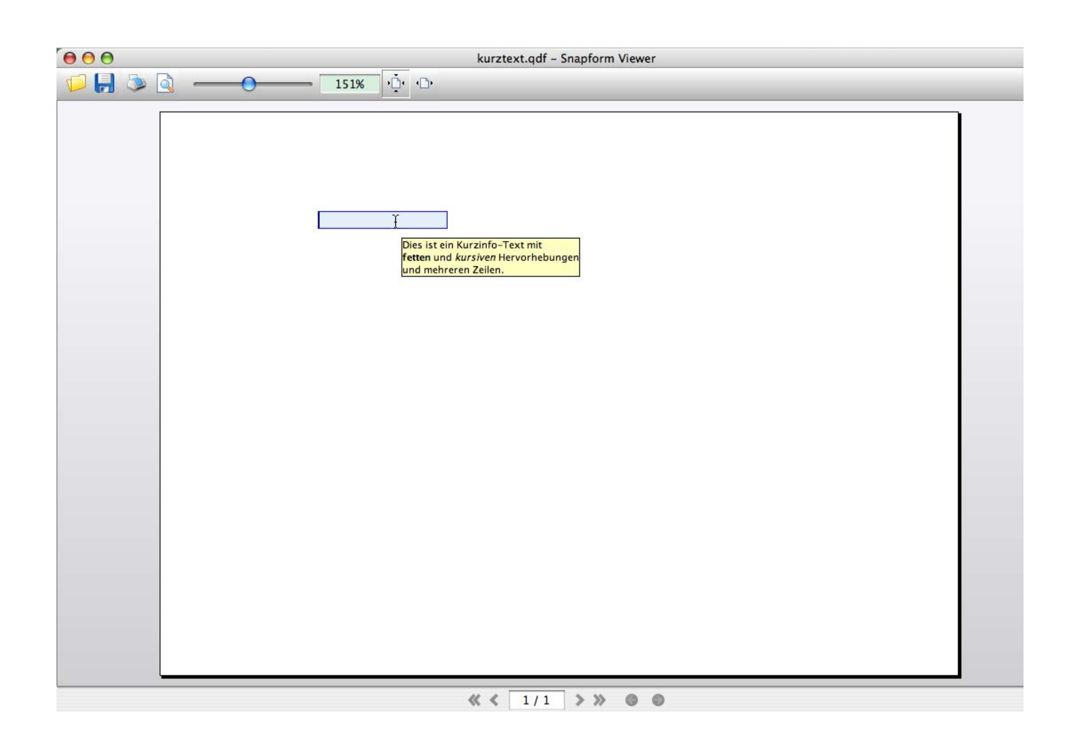
Hinweis: *Kurztexte sollten wirklich kurz sein. Daher sollte auf übermässigen Einsatz von HTML-Formatierungen, insbesonder auch auf das Einbinden von Bildern verzichtet werden.*

Fig. 4-48 *Kurzinfo-Anzeige-Fenster 044*



Der in Fig. 4-48 als HTML definierte Kurzinfo-Text erscheint im Snapform Viewer wie in Fig. 4-49 dargestellt.

Fig. 4-49 *Anzeige der Kurzinfo 045*



Ausrichtung horizontal

Diese Eigenschaft bestimmt, wie der Text im Eingabefeld ausgerichtet wird.

Wenn der Eigenschaftswert angeklickt wird, erscheint ein Drop Down Menu mit der Auswahl **links**, **zentriert**, **rechts**, welche den Text linksbündig, mittenzentriert bzw. rechtsbündig im Eingabefeld positioniert. Grundeinstellung ist **links**.

Mehrzeilige Textfelder und Passwortfelder berücksichtigen diese Eigenschaft nicht.

Eingabefelder konfigurieren

Mit dieser Eigenschaft kann ein Eingabefeld segmentiert (beispielsweise als Kammfeld) ausgebildet werden. Ein segmentiertes Feld ist ein Feld, bei welchem einzelne Zeichen oder Gruppen von Zeichen zusammengefasst und in eigenen Kästchen stehen, welche einen bestimmten Abstand voneinander haben. Diese Art Felder wird häufig für die Auswertung in Papier-basierten Arbeitsabläufen mit von Hand ausgefüllten Formularen benutzt, und sie gilt als OCR-freundlich.

Im Gegensatz zur häufig benutzten Methode, bei dieser Art von Darstellung individuelle Felder zu benutzen, erlauben segmentierte

Felder die Eingabe des Wertes "an einem Stück", was für elektronische Formulare unabdingbar ist (für weitere Berechnungen oder Datenübertragung).

Wenn für diese Eigenschaft **...** angeklickt wird, öffnet sich das Konfigurationsfenster. Dieses Fenster ist für den jeweiligen Feldtyp spezifisch und wird im betreffenden Abschnitt beschrieben.

Wenn ein Eingabefeld mit dieser Eigenschaft segmentiert wird, ändert sich das Aussehen des Felds in der Entwurfsansicht (siehe Fig. 4-50), wie auch in der Vorschau (siehe Fig. 4-51).

Fig. 4-50 *Segmentiertes Eingabefeld in der Entwurfsansicht*

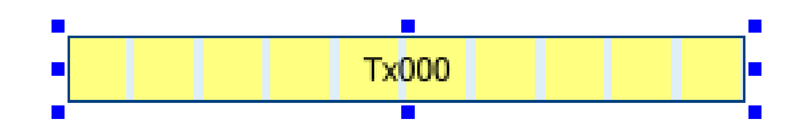
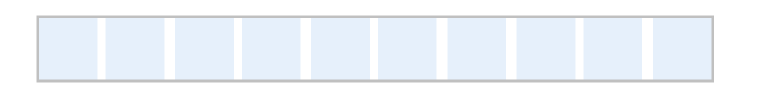


Fig. 4-51 *Segmentiertes Eingabefeld in der Vorschau*



Mehrzeilige Textfelder und Passwortfelder berücksichtigen diese Eigenschaft nicht.

Link zu Objekt

In verschiedenen Formularanwendungen gibt es Verweise auf andere Bereiche des Formulars, oder auf ergänzende Informationen. Mit der Eigenschaft **Link zu Objekt** ist es möglich, einen derartigen Verweis im Snapform-Formular abzubilden.

Wenn diese Eigenschaft gewählt wird, öffnet sich ein Drop Down-Menu mit einer Liste aller Elemente der Formularebene (alle Elemente bei welchen die Tabreihenfolge einen Wert grösser als 0 aufweist, also auch Bildfelder). Es kann nun ein Feld gewählt werden. Beim Ausfüllen des Formulars erscheint nun rechts des Feldes eine grüne Pfeilmarkierung

(siehe Fig. 4-52). Wenn diese Markierung angeklickt wird, wird das verlinkte Feld angesprungen.



Fig. 4-52 *Pfeilmarkierung bei Feld mit Link zu Objekt*

Die weiteren im Eigenschaften-Fenster zu findenden Eigenschaften sind für den jeweiligen Typ des Eingabefelds spezifisch und werden in den betreffenden Abschnitten erläutert.

4.4.2.2 Textfelder

Textfelder sind Eingabefelder für beliebigen unformatierten Text, welcher auf einer einzigen Zeile Platz findet (für mehrzeilige Textfelder siehe Abschnitt 4.4.2.3).

Das Textfeldwerkzeug erzeugt ein Feld, welches die Anzeige und Eingabe von Text erlaubt. Nach Wahl des Werkzeugs erscheint ein Cursor mit einem Textfeld in Default-Grösse (siehe Fig. 4-53).

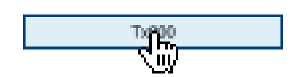
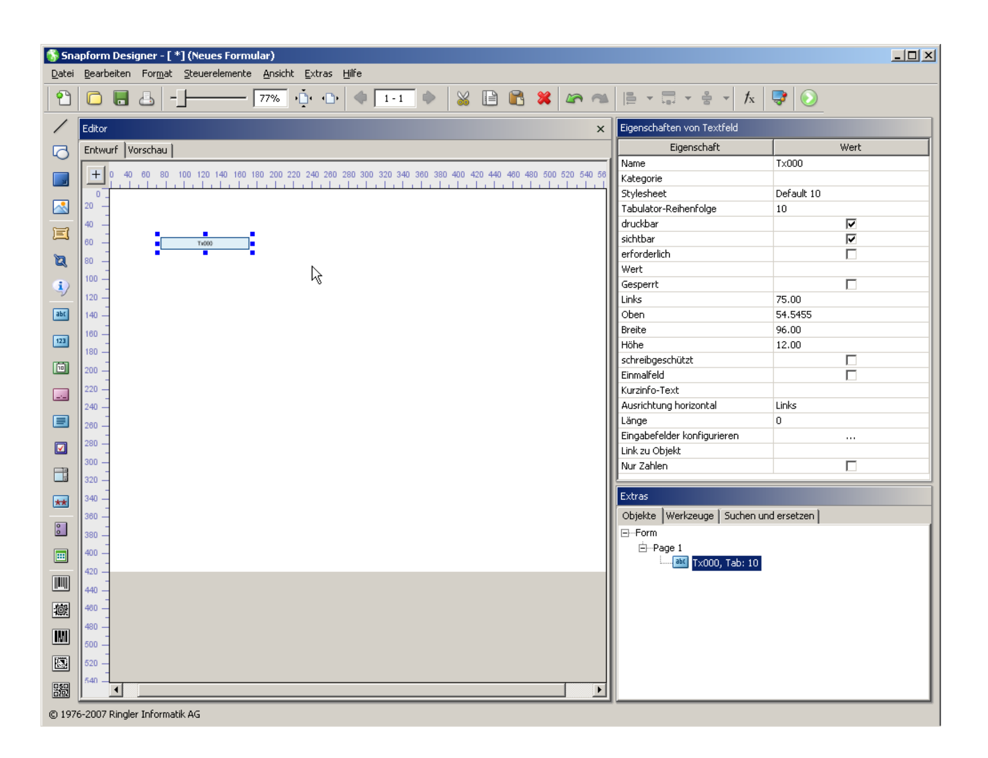


Fig. 4-53 *Textfeld-Cursor*

Mit dem Textfeld-Werkzeugt verknüpft ist ein Textfeld, 96 pt breit und 12 pt hoch. Mit Mausklick wird dieses Feld auf der Arbeitsfläche platziert. Dabei erhält es automatisch einen Namen, bestehend aus den Zeichen Tx und einer dreistelligen fortlaufenden Nummer, beginnend mit 000. Das erste in einem Formular mit dem Textfeldwerkzeug platzierte Feld trägt daher den Namen Tx000. In der Objektstruktur wird es der betreffenden Seite hinzugefügt. In Fig. 4-54 ist das erste Textfeld in einem neuen Formulars platziert, und anschliessend mit Anklicken selektiert worden.

Fig. 4-54 *Neu erstelltes Textfeld 046*



Die Länge und Breite des Textfelds kann durch Ziehen der Ankerpunkte des Selektionsrechtecks verändert werden, die Position auch durch Anfassen und Ziehen. Beim Überfahren der Ankerpunkte ändert der Cursor in Doppelpfeile, welche die Bewegungsrichtung des Ankerpunktes angeben. Beim "Anfassen" und Drücken der Maustaste ändert sich der Cursor zu einem Hohlpfeil.

Die übrigen Merkmale eines Textfeldes werden über das Eigenschaften-Fenster kontrolliert (siehe Fig. 4-55).

Fig. 4-55 *Eigenschaften eines Textfeldes 046*

Eigenschaften von Textfeld

Eigenschaft	Wert
Name	Tx000
Kategorie	
Stylesheet	Default 10
Tabulator-Reihenfolge	10
druckbar	☑
sichtbar	☑
erforderlich	☐
Wert	
Gesperrt	☐
Links	75.00
Oben	54.5455
Breite	96.00
Höhe	12.00
schreibgeschützt	☐
Einmalfeld	☐
Kurzinfo-Text	
Ausrichtung horizontal	Links
Länge	0
Eingabefelder konfigurieren	...
Link zu Objekt	
Nur Zahlen	☐

Die allgemeinen Eigenschaften von Eingabefeldern sind in Abschnitt 4.4.2.1 beschrieben. Die nachstehenden Eigenschaften sind entweder spezifisch für Textfelder oder haben Ergänzungen zu den allgemeinen Eigenschaften.

Wert

Die Eingabe von HTML-formatiertem Text ist nicht möglich.

Länge

Maximale Anzahl Zeichen im Feld. Wenn beim Ausfüllen diese Anzahl Zeichen erreicht wird, werden sämtliche weiteren eingegebenen

Zeichen ignoriert. Kein Wert oder der Wert 0 bedeutet, dass keine Längenbeschränkung herrscht.

Beim Ausfüllen wird überlanger Text durch Verkleinern der Schriftgrösse und durch Stauchen der Zeichen dargestellt, so dass der ganze Inhalt im Feld erscheint. Die Folge davon ist, dass der Text relativ schnell nicht mehr lesbar ist. Daher ist es wichtig, bereits bei der Planung des Formulars eine Grössenordnung der Textlänge zu definieren, und die Textfelder entsprechend zu dimensionieren.

Eingabefeld konfigurieren

Bei Textfeldern erfolgt die Segmentierung des Feldes in einzelne Zeichen. Diese Art segmentierter Felder wird auch "Kammfelder" genannt.

Wenn für diese Eigenschaft **...** angeklickt wird, öffnet sich das Konfigurationsfenster (siehe Fig. 4-56).

Fig. 4-56 *Konfigurationsfenster für Feldsegmentierung 047*



Im Konfigurationsfenster für Feldsegmentierung sind folgende Eigenschaften einstellbar:

Feldsegmentierung

Mit diesem Kontrollkästchen wird die Feldsegmentierung ein und ausgeschaltet. Wenn das Kontrollkästchen angeklickt ist, wird das Feld segmentiert, wenn es ausgeschaltet ist, wird das Feld nicht segmentiert.

Abstand zwischen Zeichen

Der Abstand zwischen den Kästchen für die einzelnen Zeichen in Points. Grundwert ist 1,5 pt.

Zeichenbreite

Die Breite eines Kästchens für die einzelnen Zeichen in Points. Der Grundwert beträgt 11,3 pt.

Hinweis: *Da der Grundwert für die Zeichenbreite nicht automatisch von der im Stylesheet definierten Schriftgrösse abgeleitet wird, muss darauf geachtet werden, dass die Zeichenbreite bei grösseren Schriften ausreichend ist. Als Faustregel kann für Zahlen die Schriftgrösse in Point, bei Texteingaben das 1,2-fache der Schriftgrösse in Points genommen werden.*

Bei eingeschalteter Feldsegmentierung wird die Breite des Eingabefelds entsprechend der in der Eigenschaft **Länge** festgelegten Anzahl Zeichen neu berechnet, und das Feld entsprechend skaliert. Ist noch keine Länge bestimmt worden, oder der Wert 0, wird das Eingabefeld auf 10 Zeichen ausgerichtet. Dieser Wert wird auch in die Eigenschaft **Länge** eingetragen.

Nur Zahlen

Wenn dieses Kontrollkästchen gewählt ist, können nur Ziffern zwischen 0 und 9 eingegeben werden. Sämtliche anderen Eingaben werden ignoriert.

Der Unterschied zu einem Zahlenfeld ist der, dass vorauslaufende Nullen nicht unterdrückt werden. Diese Eigenschaft wird beispielsweise für Serienummern oder ähnliche Informationen eingesetzt.

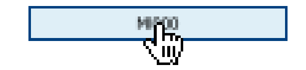
4.4.2.3 Mehrzeilige Felder

Zum Erfassen bzw. Anzeigen von längeren Texten reicht die eine einzige Zeile des Textwerkzeugs nicht aus. In diesem Fall wird das Mehrzeilen-Textfeldwerkzeug benutzt, welches eine nicht beschränkte Anzahl Zeilen Text zulässt.

Das Mehrzeilen-Textfeldwerkzeug erzeugt ein Feld, welches die Anzeige und Eingabe von mehrzeiligem Text erlaubt. Nach Wahl des

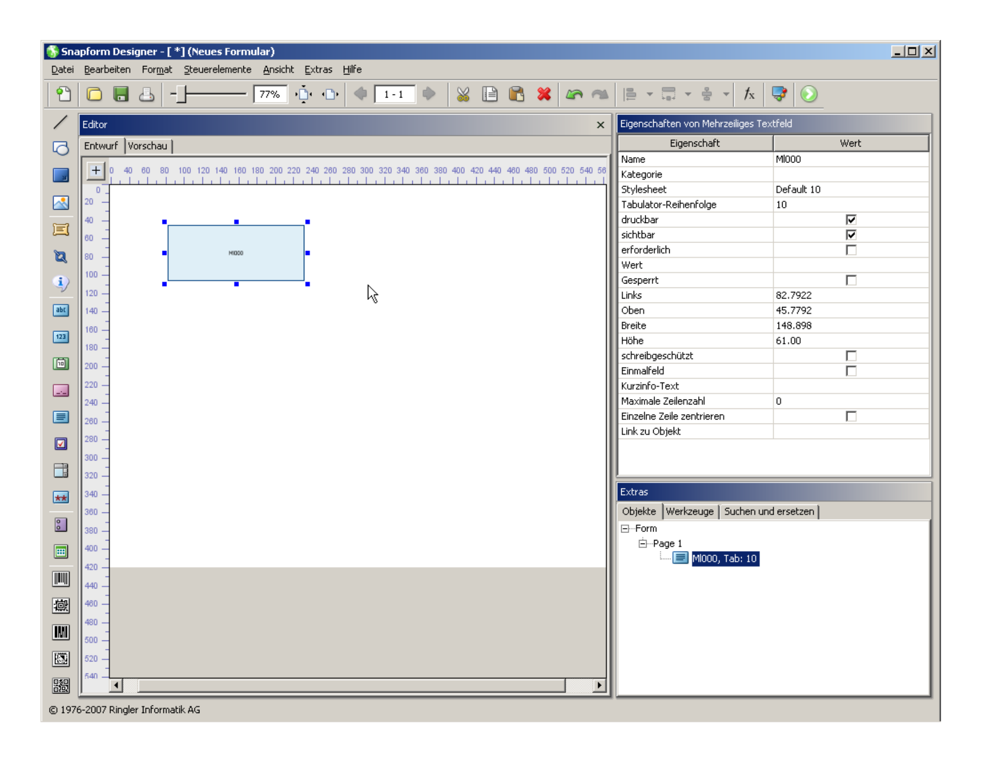
Werkzeugs erscheint ein Cursor mit einem Textfeld in Default-Grösse (siehe Fig. 4-57).

Fig. 4-57 *Mehrzeilen-Textfeld-Cursor*



Mit dem Mehrzeilen-Textfeld-Werkzeugt verknüpft ist ein Textfeld, 96 pt breit und 12 pt hoch. Mit Mausklick wird dieses Feld auf der Arbeitsfläche platziert. Dabei erhält es automatisch einen Namen, bestehend aus den Zeichen MI und einer dreistelligen fortlaufenden Nummer, beginnend mit 000. Das erste in einem Formular mit dem Mehrzeilen-Textfeldwerkzeug platzierte Feld trägt daher den Namen MI000. In der Objektstruktur wird es der betreffenden Seite hinzugefügt. In Fig. 4-58 ist das erste Textfeld in einem neuen Formulars platziert, und anschliessend mit Anklicken selektiert worden.

Fig. 4-58 *Neu erstelltes Mehrzeilen-Textfeld 048*



Die Länge und Breite des Textfelds kann durch Ziehen der Ankerpunkte des Selektionsrechtecks verändert werden, die Position auch durch Anfassen und Ziehen. Beim Überfahren der Ankerpunkte ändert der

Cursor in Doppelpfeile, welche die Bewegungsrichtung des Ankerpunktes angeben. Beim "Anfassen" und Drücken der Maustaste ändert sich der Cursor zu einem Hohlpfeil.

Die übrigen Merkmale eines Textfeldes werden über das Eigenschaften-Fenster kontrolliert (siehe Fig. 4-55).

Fig. 4-59 *Eigenschaften eines Mehrzeilen-Textfeldes 048*

Eigenschaften von Mehrzeiliges Textfeld

Eigenschaft	Wert
Name	MI000
Kategorie	
Stylesheet	Default 10
Tabulator-Reihenfolge	10
druckbar	☑
sichtbar	☑
erforderlich	☐
Wert	
Gesperrt	☐
Links	82.7922
Oben	45.7792
Breite	148.898
Höhe	61.00
schreibgeschützt	☐
Einmalfeld	☐
Kurzinfo-Text	
Maximale Zeilenzahl	0
Einzelne Zeile zentrieren	☐
Link zu Objekt	

Die allgemeinen Eigenschaften von Eingabefeldern sind in Abschnitt 4.4.2.1 beschrieben. Die nachstehenden Eigenschaften sind entweder spezifisch für Mehrzeilen-Textfelder oder haben Ergänzungen zu den allgemeinen Eigenschaften.

Wert

Die Eingabe von HTML-formatiertem Text ist nicht möglich.

Maximale Zeilenzahl

Die maximale Anzahl Zeilen, welche das Feld aufnehmen kann. Wenn der Wert 0 beträgt, besteht keine Begrenzung in der Anzahl Zeilen.

Vorsicht: *Wenn die Anzahl Zeilen nicht begrenzt ist, werden beim Erreichen des unteren Randes des Feldes die weiteren Eingaben blind angefügt. Dieser Text kann wird nicht angezeigt, und auch nicht ausgedruckt. Da keine Rollbalken in das Feld eingefügt werden, bleibt dieser Text unsichtbar. Bei der Datenübermittlung wird er übertragen. Um sicher zu stellen, dass sämtlicher Text angezeigt wird, muss daher die Anzahl Zeilen begrenzt werden.*

Hinweis: *Ein mehrzeiliges Textfeld führt keinen automatischen Zeilenumbruch durch. Daher wird beim Eingeben von langem Text die Schrift auf die gleiche Art und Weise skaliert wie bei einfachen Textfeldern. Es ist daher darauf zu achten, dass bei der Eingabe immer wieder neue Zeilen begonnen werden.*

Einzelne Zeile zentrieren

Diese Eigenschaft kontrolliert das Verhalten des Felds wenn nur eine einzige Zeile Text eingegeben ist. Normalerweise und auch bei der Eingabe ist der Text auf den oberen Rand des Feldes ausgerichtet. Wenn dieses Kontrollkästchen gewählt ist, wird hingegen der Text, wenn er nur aus einer Zeile besteht, auf die Feldmitte zentriert. Dies erlaubt, bei Feldern mit geringen Zeilenzahlen, insbesondere in Tabellen, eine harmonischere Textdarstellung.

4.4.2.4 Numerische Felder

Zahlen können auch in einfache Textfeldern eingegeben und dargestellt werden. Bei der Eingabe bieten numerische Felder jedoch wesentlich mehr Möglichkeiten zur Kontrolle der Eingabe. Zudem zeigen numerische Felder Zahlen normalerweise rechtsbündig an, was die Arbeit bei Formularen in Tabellenform vereinfacht.

In numerische Felder eingegeben werden können nur die Ziffern von 0 bis 9, das Dezimalzeichen (entsprechend der Sprachversion von Snapform Designer und den Ländereinstellungen des Rechners sowie das Minuszeichen.

Das Dezimalzeichen wird intern als "Dezimalzeichen" gespeichert, und bei unterschiedlichen Sprachversionen entsprechend korrekt dargestellt. Ein in einer deutschen Umgebung gespeichertes Formular wird daher in einer englischen Umgebung korrekt angezeigt, und verhält sich auch korrekt.

Das Minuszeichen verhält sich wie ein Umschalter. Wann immer bei der Eingabe es gedrückt wird, es schaltet die Zahl von positiv auf negativ um und umgekehrt. Das Minuszeichen muss daher auch nicht zu Beginn der Zahl eingegeben werden. Auch das Minuszeichen wird intern als "Minuszeichen" gespeichert, und entsprechend der Umgebung angezeigt.

Zahlenfelder haben Limiten in der Anzahl Stellen, welche verarbeitet werden können. Diese Grenze liegt bei 16 signifikanten Ziffern und ist technisch bedingt (praktisch sämtliche Software hat diese Limite ebenfalls). Werden auch innerhalb von Berechnungen mehr als 16 signifikante Stellen benutzt, können sich Ungenauigkeiten in den nachfolgenden Resultaten ergeben.

Das Zahlenfeldwerkzeug erzeugt ein Feld, welches die Anzeige und Eingabe von Zahlen erlaubt. Nach Wahl des Werkzeugs erscheint ein Cursor mit einem Zahlenfeld in Default-Grösse (siehe Fig. 4-60).

Fig. 4-60 *Zahlenfeld-Cursor*



Mit dem Zahlenfeld-Werkzeugt verknüpft ist ein Zahlenfeld, 96 pt breit und 12 pt hoch. Mit Mausklick wird dieses Feld auf der Arbeitsfläche platziert. Dabei erhält es automatisch einen Namen, bestehend aus den Zeichen Nr und einer dreistelligen fortlaufenden Nummer, beginnend

mit 000. Das erste in einem Formular mit dem Zahlenfeld-Werkzeug platzierte Feld trägt daher den Namen Nr000. In der Objektstruktur wird es der betreffenden Seite hinzugefügt. In Fig. 4-61 ist das erste Zahlenfeld in einem neuen Formulars platziert, und anschliessend mit Anklicken selektiert worden.

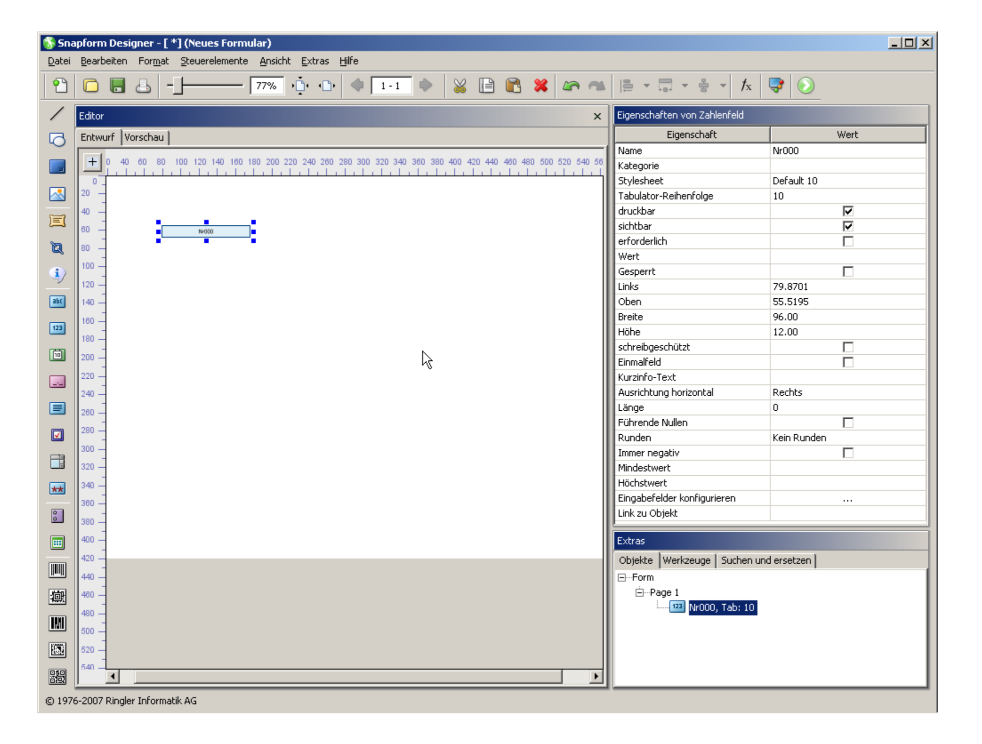


Fig. 4-61 *Neu erstelltes Zahlenfeld 049*

Die Länge und Breite des Zahlenfelds kann durch Ziehen der Ankerpunkte des Selektionsrechtecks verändert werden, die Position auch durch Anfassen und Ziehen. Beim Überfahren der Ankerpunkte ändert der Cursor in Doppelpfeile, welche die Bewegungsrichtung des Ankerpunktes angeben. Beim "Anfassen" und Drücken der Maustaste ändert sich der Cursor zu einem Hohlpfeil.

Die übrigen Merkmale eines Zahlenfelds werden über das Eigenschaften-Fenster kontrolliert (siehe Fig. 4-62).

Fig. 4-62 *Eigenschaften eines Zahlenfelds 049*

Eigenschaften von Zahlenfeld

Eigenschaft	Wert
Name	Nr000
Kategorie	
Stylesheet	Default 10
Tabulator-Reihenfolge	10
druckbar	☑
sichtbar	☑
erforderlich	☐
Wert	
Gesperrt	☐
Links	79.8701
Oben	55.5195
Breite	96.00
Höhe	12.00
schreibgeschützt	☐
Einmalfeld	☐
Kurzinfo-Text	
Ausrichtung horizontal	Rechts
Länge	0
Führende Nullen	☐
Runden	Kein Runden
Immer negativ	☐
Mindestwert	
Höchstwert	
Eingabefelder konfigurieren	...
Link zu Objekt	

Die allgemeinen Eigenschaften von Eingabefeldern sind in Abschnitt 4.4.2.1 beschrieben. Die nachstehenden Eigenschaften sind entweder spezifisch für Zahlenfelder oder haben Ergänzungen zu den allgemeinen Eigenschaften.

Stylesheet

Die Formatierung und Darstellung der Zahlen wird über die Stilvorlage gesteuert (siehe Abschnitt 4.7.1).

Praxistip: *Bei unerwarteten Resultaten der Zahlendarstellung kann eine Kontrolle der Stilvorlage im Stilvorlagen-Editor Klarheit verschaffen.*

Ausrichtung horizontal

Grundeinstellung bei der horizontalen Ausrichtung ist **rechts**. Dies erleichtert die Darstellung von mehreren Zahlen in einer Spalte.

Länge

Maximale Anzahl Zeichen im Feld. Wenn beim Ausfüllen diese Anzahl Zeichen erreicht wird, werden sämtliche weiteren eingegebenen Zeichen ignoriert. Kein Wert oder der Wert 0 bedeutet, dass keine Längenbeschränkung herrscht.

Hinweis: *Snapform hat, wie praktisch alle andere Software auch, eine technisch bedingte Limite bei der Grösse von Zahlen. Diese Limite liegt bei 16 signifikanten Stellen. Zahlen mit mehr als 16 Ziffern werden gerundet oder abgeschnitten.*

Es empfiehlt sich daher, wenn die Gefahr für zu viele verwendete Stellen besteht, die Eingabefelder auf eine geringere Anzahl Stellen zu begrenzen. Zudem sollte für die Eingabe von aus Ziffern bestehenden Zeichenketten, welche nicht als Zahl gelten (Seriennummern oder ähnlich) Textfelder mit gewählter **nur Zahlen** *Option benutzt werden.*

Beim Ausfüllen wird überlanger Text durch Verkleinern der Schriftgrösse und durch Stauchen der Zeichen dargestellt, so dass der ganze Inhalt im Feld erscheint. Die Folge davon ist, dass der Text relativ schnell nicht mehr lesbar ist. Daher ist es wichtig, bereits bei der Planung des Formulars eine Grössenordnung der Textlänge zu definieren, und die Textfelder entsprechend zu dimensionieren.

führende Nullen

Wenn dieses Kontrollkästchen gewählt ist, werden eingegebene führende Nullen nicht verworfen. Dies ist insbesondere bei Serienummern wichtig.

Gleichzeitig mit dem Wählen des Kontrollkästchens wird die der Wert von **Länge**, wenn er 0 war, auf 8 gesetzt, und **Runden** erhält den Wert **Ganzzahl**.

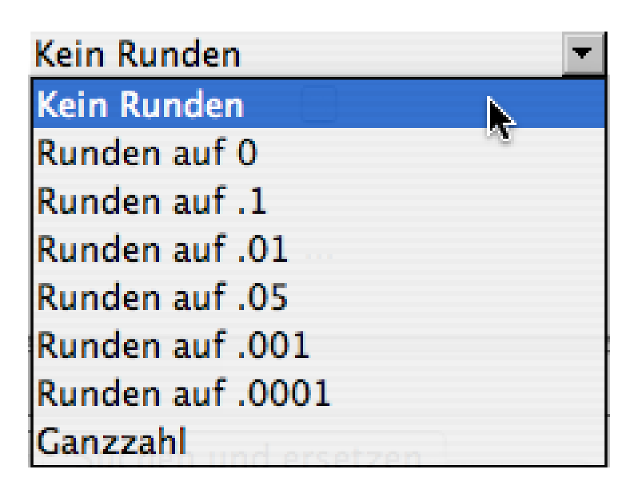
Hinweis: *Wenn dieses Kontrollkästchen unbeabsichtigt gewählt wird, müssen die beiden erwähnten Einstellungen nach dem Abwählen unbedingt kontrolliert und gegebenenfalls zurückgesetzt werden.*

Runden

Nicht unbedingt bei Eingaben, aber bei Resultaten, können mehr Nachkommastellen entstehen als erwünscht, was die Anzeige der Zahlen stören kann. Daher ist es sinnvoll gerundete Werte anzuzeigen. In dieser Feldeigenschaft wird das Rundungsverhalten eingestellt.

Wenn diese Eigenschaft gewählt wird, öffnet sich ein Drop Down Menu mit den verschiedenen Optionen (siehe Fig. 4-63).

Fig. 4-63 *Das Runden Drop Down-Menu*



Der Grundwert ist **Kein Runden**, was bedeutet, dass eingegebene und angezeigte Zahlen nicht gerundet werden. Wenn eine der nächsten Optionen **Runden auf 0**, **Runden auf .1**, **Runden auf .01**, **Runden auf .05**, **Runden auf .001**, **Runden auf .0001** gewählt ist, werden eingegebene und angezeigte Zahlen auf den betreffenden Wert gerundet. Die Option **Ganzzahl** bedeutet, dass nur ganze Zahlen akzeptiert werden, und dass die Eingabe (und Anzeige) des Dezimalzeichens nicht möglich ist.

Hinweis: *Da Berechnungen genau erfolgen, können sich Diskrepanzen zwischen den angezeigten Berechnungsresultaten und den mit gerundeten Werten durchgeführten Berechnungen ergeben. Dies ist eine Folge der*

Fehlerfortpflanzung, und muss von Anwendung zu Anwendung in Betracht gezogen werden.

Gegebenenfalls muss die Berechnung angepasst werden, indem Zahlen in der Formel mit Hilfe von **`Round(x)`** *gerundet werden, und auch die Berechnung mit gerundeten Zahlen durchgeführt wird (wohl wissend, dass das Resultat weniger genau ist).*

Immer negativ

Wenn dieses Kontrollkästchen gewählt ist, wird die betreffende Zahl immer als negative Zahl behandelt.

Mindestwert

Wenn für diese Eigenschaft ein Wert gesetzt ist, wird der eingegebene Wert kontrolliert, und nur dann akzeptiert, wenn er grösser oder gleich wie der eingegebene Mindestwert ist. (siehe auch Anmerkungen zu **Höchstwert**).

Höchstwert

Wenn für diese Eigenschaft ein Wert gesetzt ist, wird der eingegebene Wert kontrolliert, und nur dann akzeptiert, wenn er kleiner oder gleich wie der angegebene Höchstwert ist.

Wenn der Wert ausserhalb des durch **Mindestwert** und **Höchstwert** bestimmten Bereichs liegt, erscheint eine Fehlermeldung, welche auch den zulässigen Bereich angibt, und die Eingabe wird verworfen. Ein leeres Feld (oder gelöschte Eingabe) wird nicht kontrolliert.

Vorsicht: *Wird ein Höchst- oder Mindestwert eingegeben, kann er nicht mehr entfernt werden; einzig ein Verändern ist noch möglich*

Eingabefeld konfigurieren

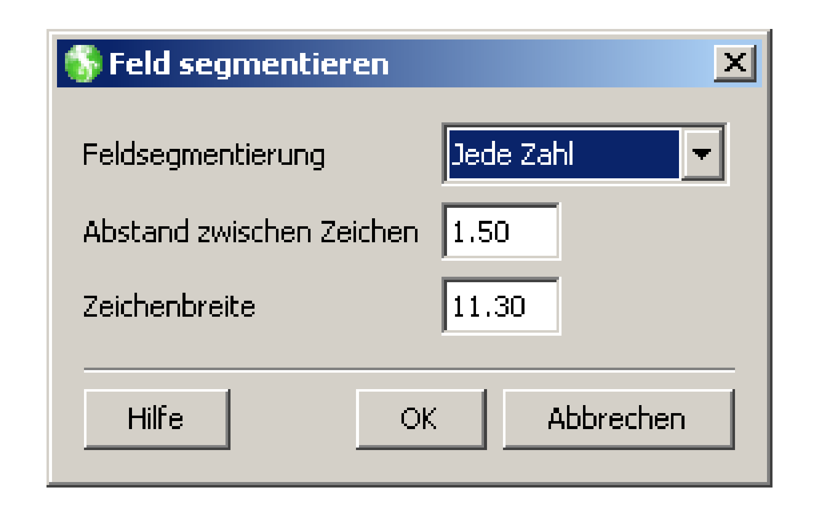
Bei Zahlenfeldern stehen verschiedene Möglichkeiten der Segmentierung zur Auswahl. erfolgt die Segmentierung des Feldes in einzelne Zeichen. Diese Art segmentierter Felder wird auch “Kammfelder” genannt.

Hinweis: *Es muss klar gestellt werden, dass die Segmentierung eine reine Darstellungssache ist. Der eigentliche Feldwert wird durch die*

Segmentierung nicht verändert (was sich bei der Art und Weise, wie er erscheint, wenn das Feld aktiv ist, klar zeigt.

Wenn für diese Eigenschaft **...** angeklickt wird, öffnet sich das Konfigurationsfenster (siehe Fig. 4-64).

Fig. 4-64 *Konfigurationsfenster für Feldsegmentierung in Zahlenfeldern 050*

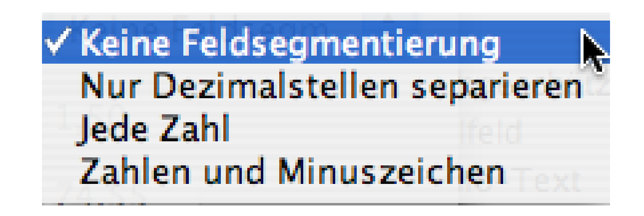


Im Konfigurationsfenster für Feldsegmentierung sind folgende Eigenschaften einstellbar:

Feldsegmentierung

Dieses Drop Down Menu steuert die Art der Segmentierung des Feldes (siehe Fig. 4-65).

Fig. 4-65 *Drop Down Menu für Feldsegmentierung in Zahlenfeldern*



Keine Feldsegmentierung

Dies ist die Grundeinstellung, und das Feld wird nicht segmentiert. Die Felder für **Abstand zwischen Zeichen** und **Zeichenbreite** sind nicht aktiv.

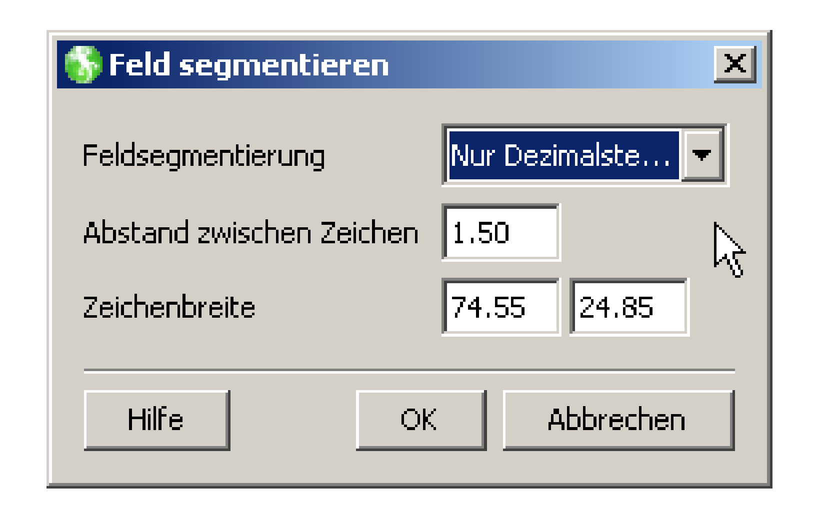
Nur Dezimalstellen separieren

In dieser Segmentierungsart wird die Zahl in einen Block vor dem Dezimalzeichen und nach dem Dezimalzeichen aufgetrennt. Das Dezimalzeichen selbst wird nicht dargestellt.

Hinweis: *Diese Segmentierungsart ist spezifisch für Währungsbeträge entwickelt worden, bei welchen eine freie Anzahl Stellen vor, und genau zwei Stellen nach dem Komma angezeigt werden. Zudem wird, ausser die Einstellung der Rundung sei .05, immer wie mit der Einstellung .01 gerundet, auch wenn eine andere Rundung (oder überhaupt keine) eingestellt ist.*

Mit Wahl dieser Segmentierungsart ändert sich das Konfigurationsfenster (siehe Fig. 4-66).

Fig. 4-66 *Konfigurationsfenster für Feldsegmentierung mit Separierung der Dezimalstellen 051*



Für weitere Erläuterungen, siehe unten:

Jede Zahl

Mit dieser Einstellung wird das Feld wie ein normales Kammfeld behandelt, wobei das Dezimalzeichen und ein allfälliges Minuszeichen nicht angezeigt werden. Das Fenster ist in Fig. 4-64 dargestellt.

Zahlen und Minuszeichen

Mit dieser Einstellung wird das Feld wie ein normales Kammfeld behandelt, wobei das Dezimalzeichen nicht angezeigt wird. Das Minuszeichen wird hingegen angezeigt. Das Fenster ist in Fig. 4-64 dargestellt.

Abstand zwischen Zeichen

Der Abstand zwischen den Kästchen für die einzelnen Zeichen in Points. Grundwert ist 1,5 pt. Dieser Wert gilt für alle Segmentierungsarten

Zeichenbreite

Die Breite eines Kästchens für die einzelnen Zeichen in Points. Der Grundwert beträgt 11,3 pt. Dieser Wert gilt für die Segmentierungsarten **Jede Zahl** und **Zahlen und Minuszeichen**. Bei der Segmentierungsart **Separierung der Dezimalstellen** ist die Breite des Bereichs vor dem Dezimalzeichen (Grundwert 74,55 pt) und die Breite des Bereichs nach dem Dezimalzeichen (Grundwert 24,85 pt) definiert.

Hinweis: *Da der Grundwert für die Zeichenbreite nicht automatisch von der im Stylesheet definierten Schriftgrösse abgeleitet wird, muss darauf geachtet werden, dass die Zeichenbreite bei grösseren Schriften ausreichend ist. Als Faustregel kann für Zahlen die Schriftgrösse in Point, bei Texteingaben das 1,2-fache der Schriftgrösse in Points genommen werden.*

Bei den Segmentierungsarten **Jede Zahl** und **Zahlen und Minuszeichen** wird die Breite des Eingabefelds entsprechend der in der Eigenschaft **Länge** festgelegten Anzahl Zeichen neu berechnet, und das Feld entsprechend skaliert. Ist noch keine Länge bestimmt worden, oder der Wert 0, wird das Eingabefeld auf 8 Zeichen ausgerichtet. Dieser Wert wird auch in die Eigenschaft **Länge** eingetragen. Bei der Segmentierungsart **Zahlen und Minuszeichen** wird ein weiteres Kästchen hinzugefügt, welches für das Minuszeichen reserviert ist. Die Segmentierungsart **Separierung der Dezimalstellen** hat hingegen keinen Einfluss auf die Anzahl Zeichen.

4.4.2.5 Datumsfelder

Datumsfelder sind spezialisierte Eingabefelder zur Eingabe von Datum und Uhrzeit. Sie erlauben es zudem, Datumsbereiche zu definieren.

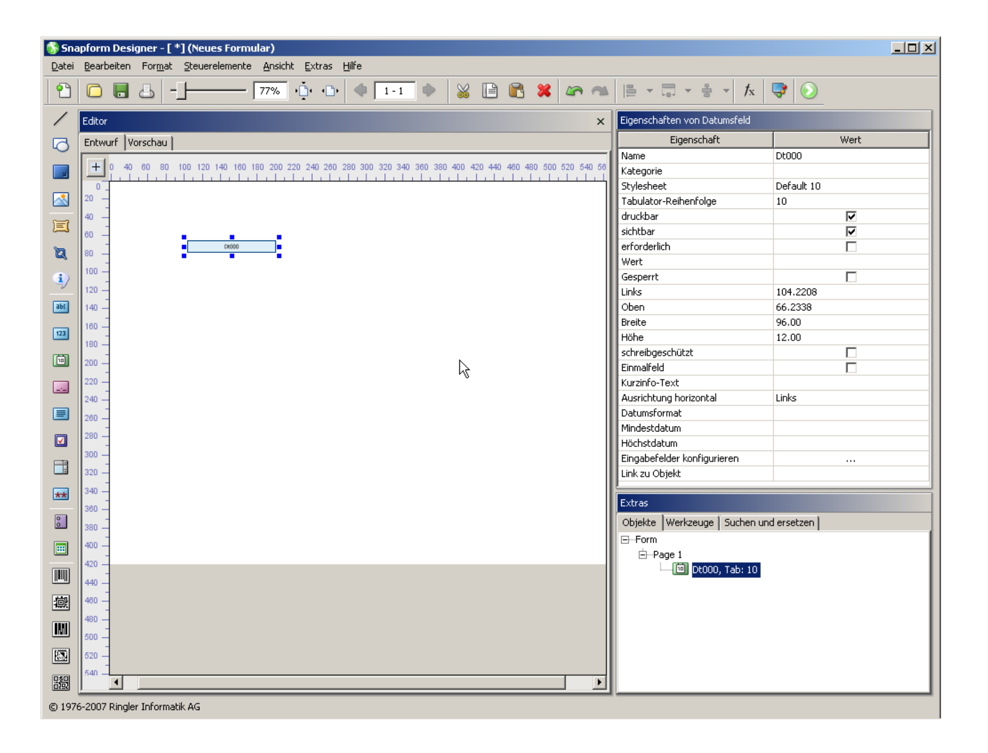
Das Datumsfeldwerkzeug erzeugt ein Feld, welches die Anzeige und Eingabe von Datum und Uhrzeit erlaubt. Nach Wahl des Werkzeugs erscheint ein Cursor mit einem Datumsfeld in Default-Grösse (siehe Fig. 4-67).

Fig. 4-67 *Datumsfeld-Cursor*



Mit dem Datumsfeld-Werkzeugt verknüpft ist ein Datumsfeld, 96 pt breit und 12 pt hoch. Mit Mausklick wird dieses Feld auf der Arbeitsfläche platziert. Dabei erhält es automatisch einen Namen, bestehend aus den Zeichen Dt und einer dreistelligen fortlaufenden Nummer, beginnend mit 000. Das erste in einem Formular mit dem Datumsfeldwerkzeug platzierte Feld trägt daher den Namen Dt000. In der Objektstruktur wird es der betreffenden Seite hinzugefügt. In Fig. 4-68 ist das erste Datumsfeld in einem neuen Formulars platziert, und anschliessend mit Anklicken selektiert worden.

Fig. 4-68 *Neu erstelltes Datumsfeld 052*



Die Länge und Breite des Datumsfelds kann durch Ziehen der Ankerpunkte des Selektionsrechtecks verändert werden, die Position auch durch Anfassen und Ziehen. Beim Überfahren der Ankerpunkte ändert der Cursor in Doppelpfeile, welche die Bewegungsrichtung des

Ankerpunktes angeben. Beim "Anfassen" und Drücken der Maustaste ändert sich der Cursor zu einem Hohlpfeil.

Die übrigen Merkmale eines Datumsfelds werden über das Eigenschaften-Fenster kontrolliert (siehe Fig. 4-69).

Fig. 4-69 *Eigenschaften eines Datumsfelds 052*

Eigenschaften von Datumsfeld

Eigenschaft	Wert
Name	Dt000
Kategorie	
Stylesheet	Default 10
Tabulator-Reihenfolge	10
druckbar	☑
sichtbar	☑
erforderlich	☐
Wert	
Gesperrt	☐
Links	104.2208
Oben	66.2338
Breite	96.00
Höhe	12.00
schreibgeschützt	☐
Einmalfeld	☐
Kurzinfo-Text	
Ausrichtung horizontal	Links
Datumsformat	
Mindestdatum	
Höchstdatum	
Eingabefelder konfigurieren	...
Link zu Objekt	

Die allgemeinen Eigenschaften von Eingabefeldern sind in Abschnitt 4.4.2.1 beschrieben. Die nachstehenden Eigenschaften sind entweder spezifisch für Datumsfelder oder haben Ergänzungen zu den allgemeinen Eigenschaften.

Stylesheet

Das Grundformat der Datumseingabe wird über die Stilvorlage gesteuert (siehe Abschnitt 4.7.1).

Praxistip: *Bei unerwarteten Resultaten der Darstellung eines Datums kann eine Kontrolle der Stilvorlage im Stilvorlagen-Editor Klarheit verschaffen.*

Wert

Hinweis: *Damit ein als Default eingegebenes Datum auch wirklich korrekt angezeigt wird, muss es im Formeleditor im Format* **yyyyMMddHHmm** *eingegeben werden. Die Darstellung erfolgt dann so wie in der Eigenschaft* **Datumsformat** *definiert.*

Datumsformat

Über diese Eigenschaft kann das Grundformat der Stilvorlage übersteuert werden, und es kann ein abweichendes Datumsformat gesetzt werden.

Beim Anklicken der Eigenschaft öffnet sich ein Drop Down Menu, aus welchem die Formate ausgewählt werden können. Zudem ist es möglich, ein eigenes Format zu definieren, welches nachher auch im Drop Down Menu zur Auswahl steht. Die Bausteine für das Datumsformat haben folgende Bedeutung:

yy	Jahr, zweistellig (z.B. 06)
yyyy	Jahr, vierstellig (z.B. 2006)
M	Monat, ohne führende Null (z.B. 5)
MM	Monat, mit führender Null (z.B. 05)
d	Tag, ohne führende Null (z.B. 7)
dd	Tag, mit führender Null (z.B. 07)
H	Stunde, ohne führende Null (z.B. 8)
HH	Stunde, mit führender Null (z.B. 08)
mm	Minute (z.B. 15)

Als Trennzeichen sind zugelassen: "." (Punkt), "/" (Schrägstrich), " " (Leerzeichen), ":" (Doppelpunkt).

Mindestdatum

Wenn für diese Eigenschaft ein Wert gesetzt ist, wird der eingegebene Wert kontrolliert, und nur dann akzeptiert, wenn das eingegebene Datum später als oder gleich wie das Mindestdatum ist. (siehe auch Anmerkungen zu **Höchstdatum**).

Höchstdatum

Wenn für diese Eigenschaft ein Wert gesetzt ist, wird der eingegebene Wert kontrolliert, und nur dann akzeptiert, wenn das eingegebene Datum früher als oder gleich wie das Höchstdatum ist.

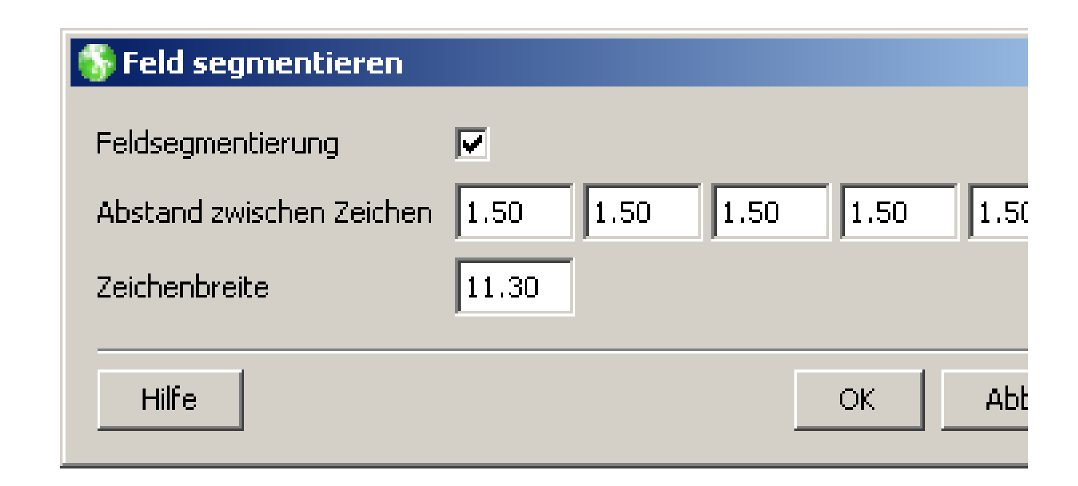
Wenn der Wert ausserhalb des durch **Mindestdatum** und **Höchstdatum** bestimmten Bereichs liegt, erscheint eine Fehlermeldung, welche auch den zulässigen Bereich angibt, und die Eingabe wird verworfen. Ein leeres Feld (oder gelöschte Eingabe) wird nicht kontrolliert.

Eingabefeld konfigurieren

Bei Datumsfeldern erfolgt die Segmentierung immer für jede einzelne Ziffer. Diese Art segmentierter Felder wird auch "Kammfelder" genannt.

Wenn für diese Eigenschaft **...** angeklickt wird, öffnet sich das Konfigurationsfenster (siehe Fig. 4-70).

Fig. 4-70 *Konfigurationsfenster für Feldsegmentierung in Datumsfeld 053*



Im Konfigurationsfenster für Feldsegmentierung sind folgende Eigenschaften einstellbar:

Feldsegmentierung

Mit diesem Kontrollkästchen wird die Feldsegmentierung ein- und ausgeschaltet. Wenn das Kontrollkästchen angeklickt ist, wird das Feld segmentiert, wenn es ausgeschaltet ist, wird das Feld nicht segmentiert.

Abstand zwischen Zeichen

Der Abstand zwischen den Kästchen für die einzelnen Zeichen in Points. Grundwert ist 1,5 pt. Bei Datumsfeldern kann der Abstand zwischen den Kästchen individuell eingestellt werden. Die Anzahl Felder für Zwischenräume richtet sich dabei nach dem gewählten Datumsformat (für das Format **d.M.yy** stehen 3 Felder zur Verfügung; für das Format **dd.MM.yyyy**, hingegen 7 Felder.

Praxistip: *Es empfiehlt sich, die Einstellungen bei starkem Zoomfaktor zu kontrollieren.*

Zeichenbreite

Die Breite eines Kästchens für die einzelnen Zeichen in Points. Der Grundwert beträgt 11,3 pt.

4.4.2.6 Eingabemasken

Eingaben müssen häufig in einem ganz bestimmten Format erfolgen, damit sie gültig sind. Ein einfaches Beispiel ist ein bestimmtes vorgegebenes Datumsformat (welches über ein Datumsfeld abgedeckt ist). Eine derartige Eingabe muss einer “Maske” entsprechen. Das zu diesem Zweck benutzte Eingabefeld ist das Maskenfeld.

Maskenfelder prüfen die Eingaben entsprechend der definierten Maske, und akzeptieren nur für die bestimmte Position in der Eingabe gültige Zeichen, und wandeln sie gegebenenfalls um. Sie können auch fixe Zeichen enthalten, welche automatisch eingefügt werden.

Wenn beim Ausfüllen des Formulars ein Maskenfeld aktiv wird, wird die Maske als Grundwert angezeigt, Für die einzugebenden Stellen wird ein

Platzhalter angezeigt. Beim Eingeben werden die Platzhalter Zeichen für Zeichen durch das eingegebene Zeichen ersetzt.

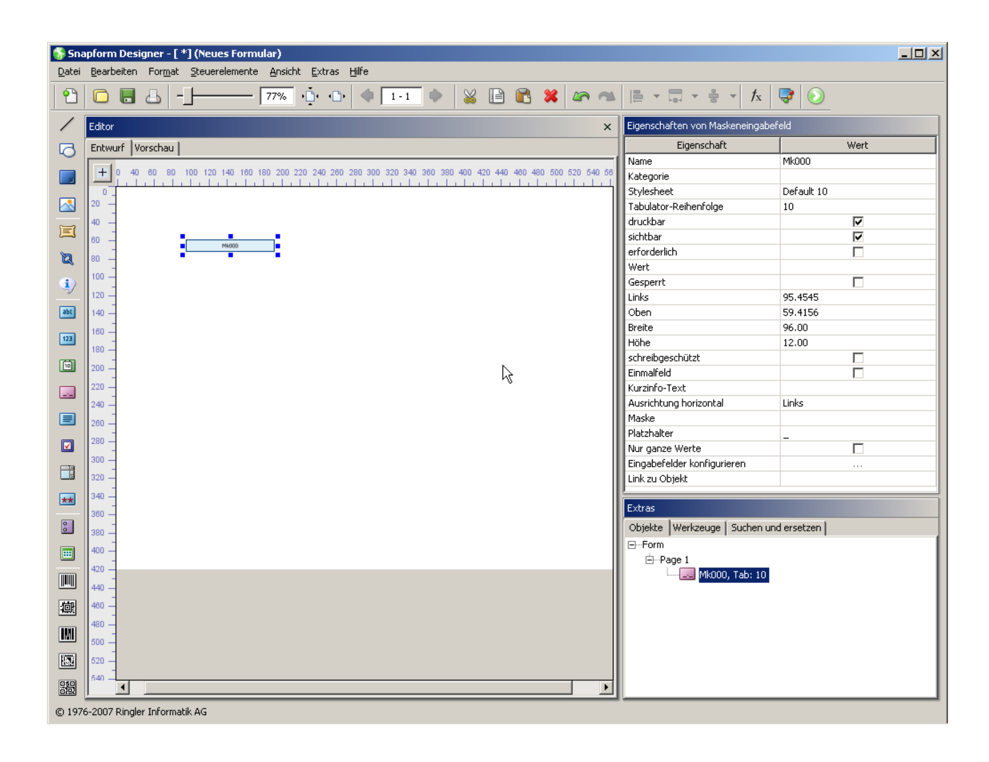
Das Maskenfeldwerkzeug erzeugt ein Feld, welches die Anzeige und Eingabe von strukturierten Zeichenketten erlaubt. Nach Wahl des Werkzeugs erscheint ein Cursor mit einem Maskenfeld in Default-Grösse (siehe Fig. 4-71).

Fig. 4-71 *Maskenfeld-Cursor*



Mit dem Maskenfeld-Werkzeugt verknüpft ist ein Maskenfeld, 96 pt breit und 12 pt hoch. Mit Mausklick wird dieses Feld auf der Arbeitsfläche platziert. Dabei erhält es automatisch einen Namen, bestehend aus den Zeichen Mk und einer dreistelligen fortlaufenden Nummer, beginnend mit 000. Das erste in einem Formular mit dem Maskenfeld-Werkzeug platzierte Feld trägt daher den Namen Mk000. In der Objektstruktur wird es der betreffenden Seite hinzugefügt. In Fig. 4-72 ist das erste Maskenfeld in einem neuen Formular platziert, und anschliessend mit Anklicken selektiert worden.

Fig. 4-72 *Neu erstelltes Maskenfeld 054*



Die Länge und Breite des Maskenfelds kann durch Ziehen der Ankerpunkte des Selektionsrechtecks verändert werden, die Position auch durch Anfassen und Ziehen. Beim Überfahren der Ankerpunkte ändert der Cursor in Doppelpfeile, welche die Bewegungsrichtung des Ankerpunktes angeben. Beim "Anfassen" und Drücken der Maustaste ändert sich der Cursor zu einem Hohlpfeil.

Die übrigen Merkmale eines Maskenfelds werden über das Eigenschaften-Fenster kontrolliert (siehe Fig. 4-73).

Fig. 4-73 *Eigenschaften eines Maskenfelds 054*

Eigenschaften von Maskeneingabefeld

Eigenschaft	Wert
Name	Mk000
Kategorie	
Stylesheet	Default 10
Tabulator-Reihenfolge	10
druckbar	☑
sichtbar	☑
erforderlich	☐
Wert	
Gesperrt	☐
Links	95.4545
Oben	59.4156
Breite	96.00
Höhe	12.00
schreibgeschützt	☐
Einmalfeld	☐
Kurzinfo-Text	
Ausrichtung horizontal	Links
Maske	
Platzhalter	_
Nur ganze Werte	☐
Eingabefelder konfigurieren	...
Link zu Objekt	

Die allgemeinen Eigenschaften von Eingabefeldern sind in Abschnitt 4.4.2.1 beschrieben. Die nachstehenden Eigenschaften sind entweder spezifisch für Maskenfelder oder haben Ergänzungen zu den allgemeinen Eigenschaften.

Stylesheet

Die Formatierung und Darstellung der Eingabe und des Erscheinungsbildes des Felds wird über die Stilvorlage gesteuert (siehe Abschnitt 4.7.1).

Maske

Das Kernelement des Maskenfeldes ist die Maske. Über die Maske wird die Struktur des einzugebenden und anzuzeigenden Werts gesteuert. Die Maske ist eine Zeichenkette, welche bestimmte Maskenzeichen sowie übrige konstante Zeichen umfasst. Die folgenden Maskenzeichen stehen zur Verfügung:

Zahl

U Grossbuchstabe; Kleinbuchstaben werden automatisch gross geschrieben.

L Kleinbuchstabe; Grossbuchstaben werden automatisch klein geschrieben.

A Buchstabe oder Zahl

? Buchstabe (ohne Kontrolle der Gross- oder Kleinschreibung)

***** Jedes Zeichen

H Hexadezimal-Zeichen (0-9, a-f oder A-F)

' (Apostroph), Schutz-Zeichen; wird benutzt, um ein Steuerzeichen zu "schützen", wenn es in der Eingabe als solches auftauchen soll.

Beispiele für Eingabemasken befinden sich im Musterdokument masks.qdf.

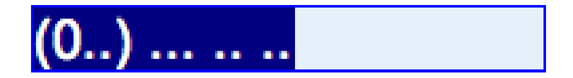
Wenn ein Maskenzeichen literal (als solches) in einer Maske auftreten soll, wird es mit einem Apostroph geschützt (der Apostroph ist das Escape-Zeichen).

Platzhalte

Beim Ausfüllen des Maskenfelds wird die Maske selbst dargestellt. Dabei werden konstante Zeichen als solche angezeigt, und für die einzugebenden Zeichen wird ein Platzhalter benutzt. Dieser Platzhalter wird in dieser Eigenschaft definiert. Normalerweise ist es der Unterstrich ("_"), aber es können beliebige Zeichen (z.B. Punkt, Leerzeichen, Bullet) verwendet werden.

Die Maske **(0##) ### ## ##** sieht in der Vorschau wie in Fig. 4-74 gezeigt aus. Zur besseren Darstellung wurde der Punkt als Platzhalter gewählt.

Fig. 4-74 *Aktives Maskenfeld in der Vorschau*



Nur ganze Werte

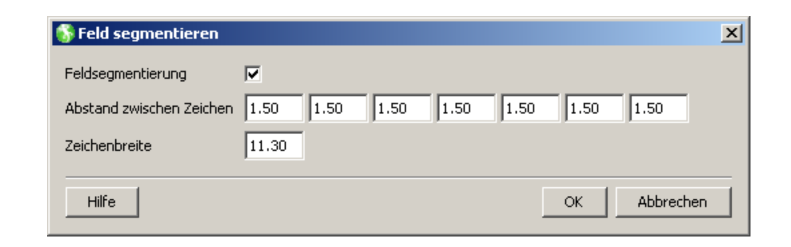
Wenn dieses Kontrollkästchen gewählt ist, wird eine Eingabe nur dann übernommen, wenn sie vollständig ist, was bedeutet, dass sämtliche Platzhalter durch eine Eingabe ersetzt worden sind. Eine unvollständige Eingabe wird verworfen.

Eingabefeld konfigurieren

Bei Maskenfeldern erfolgt die Segmentierung immer für jedes einzelne Zeichen, einschliesslich der konstanten Zeichen der Maske. Diese Art segmentierter Felder wird auch "Kammfelder" genannt.

Wenn für diese Eigenschaft **...** angeklickt wird, öffnet sich das Konfigurationsfenster (siehe Fig. 4-75).

Fig. 4-75 *Konfigurationsfenster für Feldsegmentierung in Maskenfeld 055*



Im Konfigurationsfenster für Feldsegmentierung sind folgende Eigenschaften einstellbar:

Feldsegmentierung

Mit diesem Kontrollkästchen wird die Feldsegmentierung ein- und ausgeschaltet. Wenn das Kontrollkästchen angeklickt ist, wird das Feld segmentiert, wenn es ausgeschaltet ist, wird das Feld nicht segmentiert.

Abstand zwischen Zeichen

Der Abstand zwischen den Kästchen für die einzelnen Zeichen in Points. Grundwert ist 1,5 pt. Bei Maskenfeldern kann der Abstand zwischen den

Kästchen individuell eingestellt werden. Die Anzahl Felder für Zwischenräume richtet sich dabei nach der Maske.

Praxistip: *Es empfiehlt sich, die Einstellungen bei starkem Zoomfaktor zu kontrollieren.*

Zeichenbreite

Die Breite eines Kästchens für die einzelnen Zeichen in Points. Der Grundwert beträgt 11,3 pt.

Die Maske für das Beispiel in Fig. 4-75 lautet **(0##)#######**. Da die Zwischenräume einzeln einstellbar sind, braucht die Maske selbst keine Leerzeichen als Zwischenräume zu enthalten. Das resultierende Feld in der Entwurfsansicht ist in Fig. 4-76 gezeigt, und ausgefüllt, in der Vorschau in Fig. 4-77.

Fig. 4-76 *Segmentiertes Eingabefeld in der Entwurfsansicht*

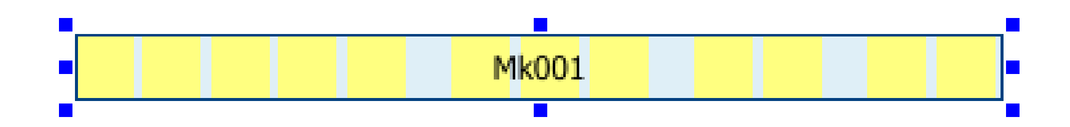


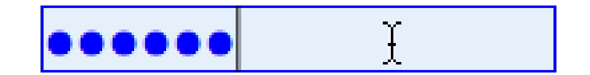
Fig. 4-77 *Segmentiertes Eingabefeld in der Vorschau*



4.4.2.7 Passwortfelder

Es gibt Eingaben, welche nicht im Klartext angezeigt werden dürfen, beispielsweise Passwörter. Für diese Art Eingaben werden Passwortfelder eingesetzt. Passwortfelder sind im Wesentlichen Textfelder, bei welchen die Anzeige der Eingabe unterdrückt wird. Anstelle des eingegebenen Zeichens wird ein grosser Punkt angezeigt. In Fig. 4-78 ist ein Passwortfeld mit Eingabe in der Vorschau gezeigt.

Fig. 4-78 *Passwortfeld mit Eingabe in der Vorschau*



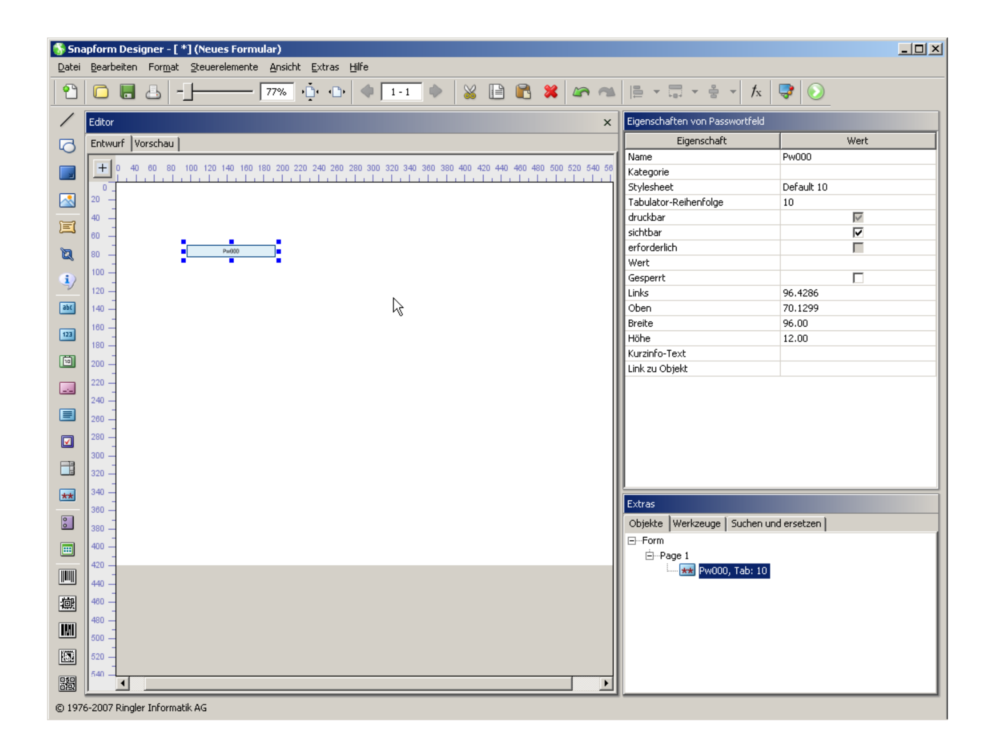
Das Passwortfeldwerkzeug erzeugt ein Feld, welches die Eingabe in Klartext unterdrückt und statt dessen ein Ersatzzeichen benutzt. Nach Wahl des Werkzeugs erscheint ein Cursor mit einem Passwortfeld in Default-Grösse (siehe Fig. 4-79).

Fig. 4-79 *Passwortfeld-Cursor*



Mit dem Passwortfeld-Werkzeugt verknüpft ist ein Passwortfeld, 96 pt breit und 12 pt hoch. Mit Mausklick wird dieses Feld auf der Arbeitsfläche platziert. Dabei erhält es automatisch einen Namen, bestehend aus den Zeichen Pw und einer dreistelligen fortlaufenden Nummer, beginnend mit 000. Das erste in einem Formular mit dem Passwortfeld-Werkzeug platzierte Feld trägt daher den Namen Pw000. In der Objektstruktur wird es der betreffenden Seite hinzugefügt. In Fig. 4-80 ist das erste Passwortfeld in einem neuen Formular platziert, und anschliessend mit Anklicken selektiert worden.

Fig. 4-80 *Neu erstelltes Passwortfeld 056*



Die Länge und Breite des Passwortfelds kann durch Ziehen der Ankerpunkte des Selektionsrechtecks verändert werden, die Position auch durch Anfassen und Ziehen. Beim Überfahren der Ankerpunkte ändert der Cursor in Doppelpfeile, welche die Bewegungsrichtung des Ankerpunktes angeben. Beim "Anfassen" und Drücken der Maustaste ändert sich der Cursor zu einem Hohlpfeil.

Die übrigen Merkmale eines Passwortfelds werden über das Eigenschaften-Fenster kontrolliert (siehe Fig. 4-81).

Fig. 4-81 *Eigenschaften eines Passwortfelds 056*

Eigenschaften von Passwortfeld

Eigenschaft	Wert
Name	Pw000
Kategorie	
Stylesheet	Default 10
Tabulator-Reihenfolge	10
druckbar	☑
sichtbar	☑
erforderlich	☐
Wert	
Gesperrt	☐
Links	96.4286
Oben	70.1299
Breite	96.00
Höhe	12.00
Kurzinfo-Text	
Link zu Objekt	

Die allgemeinen Eigenschaften von Eingabefeldern sind in Abschnitt 4.4.2.1 beschrieben. Die nachstehenden Eigenschaften sind entweder spezifisch für Maskenfelder oder haben Ergänzungen zu den allgemeinen Eigenschaften.

Stylesheet

Die Darstellung der Eingabe und des Erscheinungsbildes des Felds wird über die Stilvorlage gesteuert (siehe Abschnitt 4.7.1).

druckbar

Ein Passwortfeld ist nicht druckbar, und kann nicht druckbar gemacht werden.

Wert

Es kann zwar ein Grundwert eingegeben werden, was aber wenig sinnvoll ist, da er einfach durch eine Reihe von Punkten dargestellt wird. Hingegen kann eine Auswerteformel für die Passworteingabe hinterlegt werden.

4.4.2.8 Eingabefelder-Beispiele

Das Dokument eingabefelder.qdf (siehe Fig. 4-82 (Entwurfsmodus)) bzw. Fig. 4-83 (Ansicht in Snapform Viewer, ausgefüllt)) sind verschiedene Beispiele von Eingabefeldern, wie in den vorangehenden Abschnitten beschrieben, gezeigt.

Nach dem Selektieren der Feldelemente im Snapform Designer gibt das Eigenschaften-Fenster Auskunft über dessen jeweiligen Merkmale. Ein Doppelklick auf das Feld öffnet den Formeleditor zur Anzeige des Texts.

Fig. 4-82 *Eingabefelder-Beispiele von eingabefelder.qdf, im Entwurfs-Modus 057*

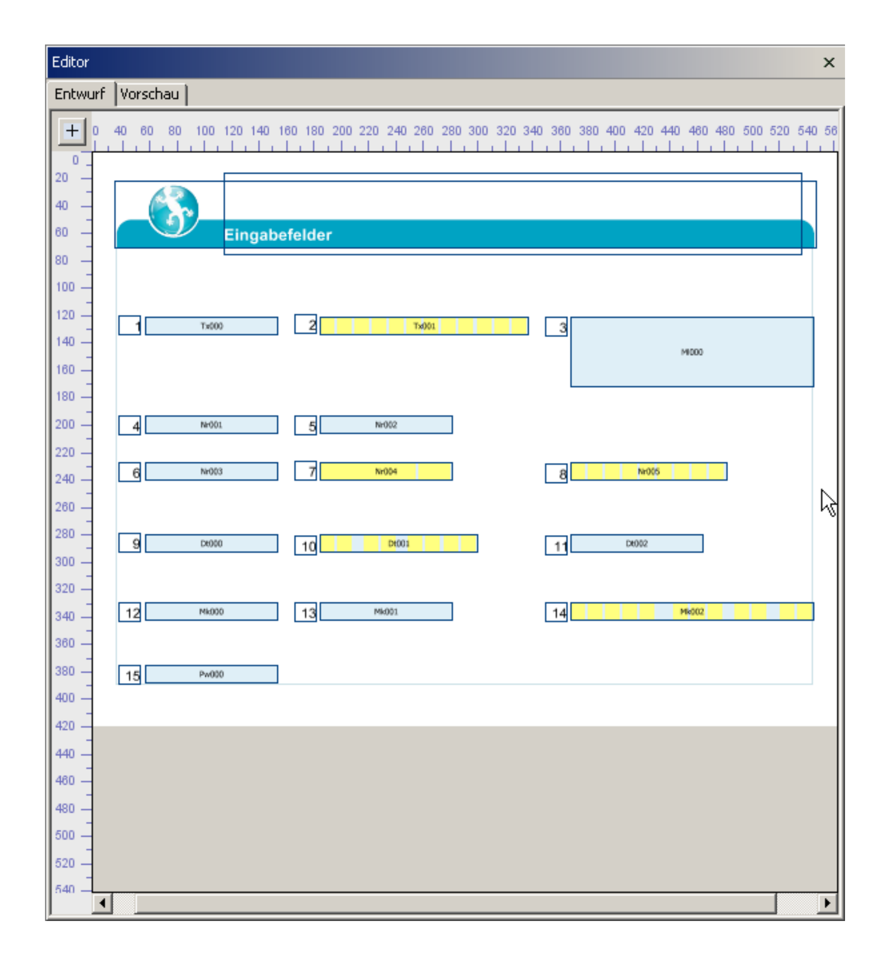
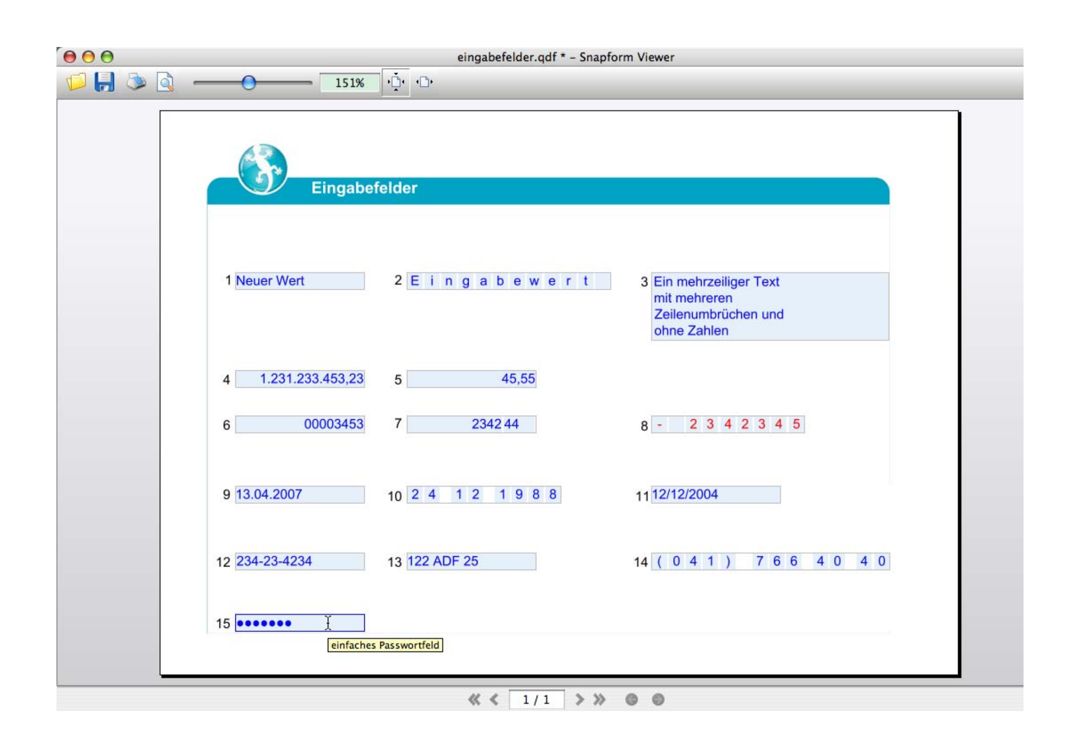


Fig. 4-83 *Ausgefüllte Eingabefelder-Beispiele von eingabefelder.qdf in Snapform Viewer 058*



1 Tx000: Einfaches Textfeld mit Grundwert
2 Tx001: Segmentiertes Textfeld
3 Ml000: Mehrzeiliges Textfeld
4 Nr001: Zahlenfeld, keine Rundung, kein Format
5 Nr002: Zahlenfeld, Wert zwischen 10 und 150, Runden auf .05
6 Nr003: Zahlenfeld mit führenden Nullen, 8 Stellen
7 Nr004: segmentiertes Zahlenfeld mit separiertem Dezimalzeichen und Runden auf .001
8 Nr005: Segmentiertes Zahlenfeld, Zahlen und Minuszeichen, Länge 8 Zeichen und Runden auf .01
9 Dt000: einfaches Datumsfeld, Format **dd.MM.yyyy**
10 Dt001: segmentiertes Datumsfeld, Format **dd.MM.yyyy**
11 Dt002: einfaches Datumsfeld, Format **dd/MM/yyyy**; das eingegebene Datum muss im 21. Jahrhundert liegen
12 Mk000: Maskenfeld, Maske **###-##-####** (Social Security Number, US)
13 Mk001: Maskenfeld, Maske ### AAA 25 (Autonummer im französischen Departement Drôme)
14 Mk002: Maskenfeld segmentiert, Maske **(0##)#######** (schweizerische Telefonnummer); Segmentierung separiert die Nummernblöcke; Platzhalter ist **#**
15 Pw000: einfaches Passwortfeld

4.4.3 Kontrollkästchen und Optionsfelder

Ein häufiges Formularelement ist die Frage, ob ein bestimmter Sachverhalt zutrifft, oder welche Option für einen bestimmten Sachverhalt zutrifft. Diese Frage wird in Papier-Formularen über ein oder mehrere Felder zum Ankreuzen behandelt. In elektronischen Formularen geschieht dies mit Kontrollkästchen und Optionsfeldern.

Ein Kontrollkästchen ist ein Formularelement bei welchem ein bestimmter Sachverhalt als Zutreffend oder Nicht Zutreffend gekennzeichnet werden kann.

Optionsfelder sind eine Gruppe von Formularelementen, welche verschiedene sich gegenseitig ausschliessende Optionen eines bestimmten Sachverhalts kennzeichnen können, und genau ein Sachverhalt zutreffen muss.

Wie aus diesen Definitionen zu erkennen ist, sind Kontrollkästchen und Optionsfelder sehr stark miteinander verwandt. Dies zeigt sich in der Art und Weise, wie diese beiden Formularelemente in Snapform implementiert worden sind.

Das Grundelement ist das Kontrollkästchen, welches die Basis-Funktionalität bereit stellt. Für Optionsfelder wird ein "Container" benutzt, in welchem die benötigte Anzahl Kontrollkästchen platziert wird. Diese Kontrollkästchen funktionieren dadurch als Optionsfelder gemäss obiger Definition.

4.4.3.1 Kontrollkästchen

In Snapform werden Kontrollkästchen in der Grundeinstellung als Quadrate dargestellt, und das Zeichen für "gewählt" ist ein Kreuz. Dieses Zeichen ist fix und kann nicht verändert werden. Die Grösse des Zeichens ist von der Feldgrösse abhängig. Wenn jedoch das Wählen des Kontrollkästchens das Resultat einer Formel ist, wird das Kreuz etwas kleiner dargestellt.

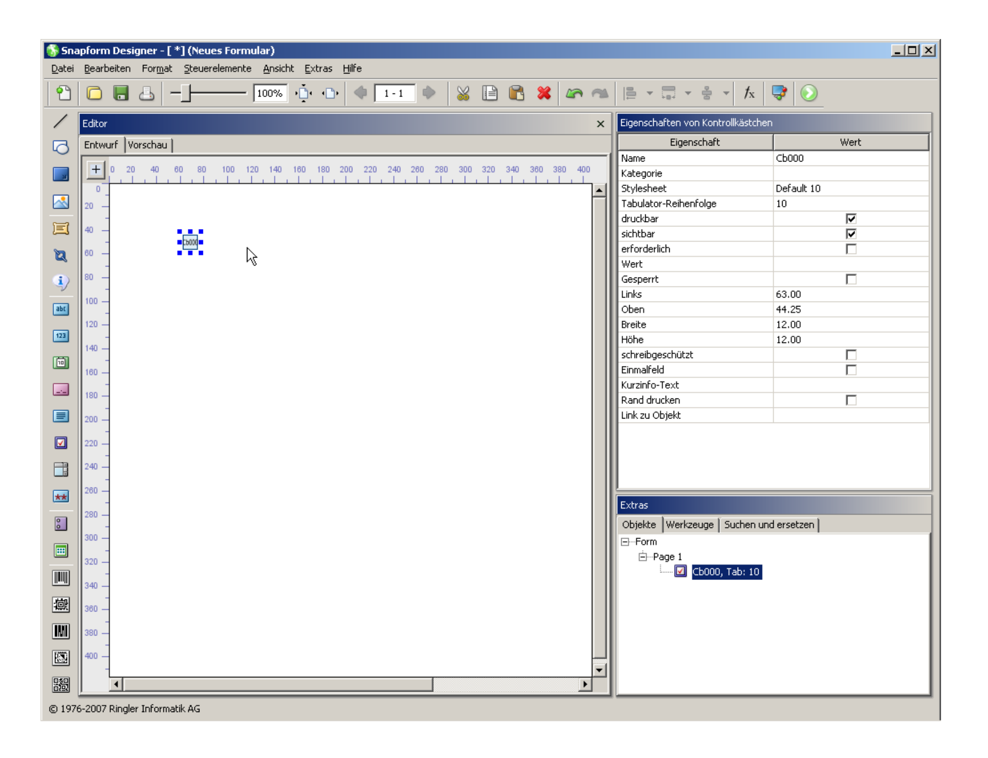
Das Kontrollkästchenfeldwerkzeug erzeugt ein Kontrollkästchen. Nach Wahl des Werkzeugs erscheint ein Cursor mit einem Kontrollkästchen in Default-Grösse (siehe Fig. 4-84).

Fig. 4-84 *Kontrollkästchen-Cursor*



Mit dem Kontrollkästchen-Werkzeugt verknüpft ist ein Kontrollkästchen-Feld, 12 pt breit und 12 pt hoch. Mit Mausklick wird dieses Feld auf der Arbeitsfläche platziert. Dabei erhält es automatisch einen Namen, bestehend aus den Zeichen Cb und einer dreistelligen fortlaufenden Nummer, beginnend mit 000. Das erste in einem Formular mit dem Kontrollkästchen-Werkzeug platzierte Feld trägt daher den Namen Cb000. In der Objektstruktur wird es der betreffenden Seite hinzugefügt. In Fig. 4-85 ist das erste Kontrollkästchen in einem neuen Formular platziert, und anschliessend mit Anklicken selektiert worden.

Fig. 4-85 *Neu erstelltes Kontrollkästchen 059*



Die Länge und Breite des Kontrollkästchen kann durch Ziehen der Ankerpunkte des Selektionsrechtecks verändert werden, die Position

auch durch Anfassen und Ziehen. Beim Überfahren der Ankerpunkte ändert der Cursor in Doppelpfeile, welche die Bewegungsrichtung des Ankerpunktes angeben. Beim "Anfassen" und Drücken der Maustaste ändert sich der Cursor zu einem Hohlpfeil.

Die übrigen Merkmale eines Kontrollkästchen werden über das Eigenschaften-Fenster kontrolliert (siehe Fig. 4-86).

Fig. 4-86 *Eigenschaften eines Kontrollkästchens 059*

Eigenschaften von Kontrollkästchen

Eigenschaft	Wert
Name	Cb000
Kategorie	
Stylesheet	Default 10
Tabulator-Reihenfolge	10
druckbar	☑
sichtbar	☑
erforderlich	☐
Wert	
Gesperrt	☐
Links	63.00
Oben	44.25
Breite	12.00
Höhe	12.00
schreibgeschützt	☐
Einmalfeld	☐
Kurzinfo-Text	
Rand drucken	☐
Link zu Objekt	

Name

Der Name des Kontrollkästchens im Dokument. Dieser Name kann geändert werden.

Kategorie

Zugehörigkeit zu einer oder mehreren Kategorien, welche es erlauben, logisch zusammengehörende Gruppen von Elementen zu bilden. Beim Anklicken dieses Felds öffnet sich der Kategorien Dialog (siehe Fig. 4-23).

Die Kategorien werden durch Anklicken des betreffenden Kontrollkästchens ausgewählt. Mit **OK** wird die Auswahl übernommen.

Stylesheet

Das für das Element gültige Stylesheet. Wenn explizite kein Stylesheet ausgewählt worden ist, wird das erste Default-Stylesheet genutzt. Für Kontrollkästchen sind vor allem die **Allgemeinen Einstellungen** von Belang.

Tabulator-Reihenfolge

Ordnungsnummer des Elements in der Tabulator-Reihenfolge. Dieser Wert wird automatisch vergeben, und liegt in der Reihe der Feld-Tabulator-Reihenfolge (welche mit 10 beginnend in Zehnerschritten erhöht wird). Das erste Feld des Formulars hat daher den Tabulator-Reihenfolge-Wert 10, das zweite den Wert 20, etc.. Der Wert kann verändert werden, so dass eine frei bestimmbare Tabulatorreihenfolge aufgebaut werden kann, welche nicht mit der Reihenfolge des Hinzufügens der Felder übereinstimmen muss.

druckbar

Wenn dieses Kontrollkästchen gewählt ist, wird das Kontrollkästchen ausgedruckt.

sichtbar

Wenn dieses Kontrollkästchen gewählt ist, wird das Kontrollkästchen auf dem Bildschirm angezeigt.

erforderlich

Wenn dieses Kontrollkästchen gewählt ist, muss dieses Kontrollkästchen mindestens einmal umgeschaltet worden sein, damit das Formular gültig ist (für Formular-Gültigkeit, siehe auch Abschnitt 4.8.4.2).

Wert

Es ist zwar möglich, bei Kontrollkästchen einen konstanten Wert zu definieren, aber wenig sinnvoll. Anstelle dessen wird hier die Formel,

welche die mit dem Umschalten des Kontrollkästchens verbundene Aktionen steuert, eingegeben.

Die Formel wird beim Umschalten des Kontrollkästchens sowie beim Öffnen des Dokuments und im Rahmen der übrigen Formelverarbeitungsläufe ausgeführt.

Gesperrt

Wenn dieses Kontrollkästchen gewählt ist, wird das Kontrollkästchen im Snapform Designer blockiert, und kann nicht (versehentlich) verschoben werden. Zudem werden sämtliche übrigen Eigenschaften im Eigenschaften-Dialog deaktiviert. Der Zugriff auf die Eigenschaften ist erst wieder möglich, wenn dieses Kontrollkästchen deaktiviert wird.

Links

X-Koordinate der linken oberen Ecke der Umrandungsbox des Elements in Points. Diese Eigenschaft erlaubt eine genaue numerische Positionierung des Kontrollkästchens.

Oben

Y-Koordinate der linken oberen Ecke der Umrandungsbox des Elements in Points. Diese Eigenschaft erlaubt eine genaue numerische Positionierung des Kontrollkästchens.

Breite

Breite der Umrandungsbox des Kontrollkästchens.

Höhe

Höhe der Umrandungsbox des Kontrollkästchens.

schreibgeschützt

Wenn dieses Kontrollkästchen gewählt ist, ist das Kontrollkästchen als schreibgeschützt markiert, und dessen Inhalt kann nicht manuell überschrieben werden. Der Wert kann nach wie vor über Formeln verändert werden. Ein schreibgeschütztes Kontrollkästchen ist nach wie vor in der Tab-Reihenfolge eingebunden, und dessen Formeln werden auch entsprechend des Zustands ausgeführt.

Einmalfeld

Es gibt Informationen, welche nach dem ursprünglichen Ausfüllen des Formulars nicht mehr verändert werden dürfen und daher geschützt

werden müssen. In Snapform erfolgt dieser Schutz mit Einmalfeldern, welche beim Ausfüllen bis zum ersten Speichern verändert werden können, anschliessend aber als schreibgeschützt markiert werden.

Wenn dieses Kontrollkästchen gewählt ist, kann das Kontrollkästchen bis zum ersten Speichern eines ausgefüllten Formulars umgeschaltet werden. Nach dem ersten Speichern des Formulars wird das Kontrollkästchen als schreibgeschützt markiert.

Kurzinfo-Text

Die Kurzinfo ist ein Hilfetext, welcher erscheint, wenn der Mauscursor während einer gewissen Zeit über dem Feld liegt. Dieser Text ist in vielen Fällen die erste und einfachste Hilfe zum betreffenden Feld. Mit zunehmenden Anforderungen an die Barrierefreiheit von elektronischen Formularen erhält der Kurzinfo-Text eine zunehmend grössere Bedeutung, da Screenreader diese Information benutzen. Es ist daher, besonders für Formulare im Behörden-Umfeld, wichtig, einen gut abgefassten Kurzinfo-Text einzufügen.

Wenn diese Eigenschaft gewählt wird, öffnet sich das Kurzinfo-Anzeigefenster (siehe Fig. 4-48). Wie der Erläuterungstext angibt, können für die Kurzinfos auch HTML-Texte eingegeben werden. Es gelten hierfür die gleichen Bedingungen wie für Beschreibungsfelder. Für weitere Anmerkungen zu Kurztexten siehe deren Beschreibung im Abschnitt 4.4.2.1.

Hinweis: *Kurztexte sollten wirklich kurz sein. Daher sollte auf übermässigen Einsatz von HTML-Formatierungen, insbesonder auch auf das Einbinden von Bildern verzichtet werden.*

Rand drucken

Wenn dieses Kontrollkästchen gewählt ist, wird der Rand des Kontrollkästchenfelds ausgedruckt, sonst nicht. Diese Einstellmöglichkeit ist bei Formularen ohne Hintergrundbild sehr sinnvoll, denn sonst ist es nicht möglich, ein Kontrollkästchen im Ausdruck zu erkennen.

Link zu Objekt

In verschiedenen Formularanwendungen gibt es Verweise auf andere Bereiche des Formulars, oder auf ergänzende Informationen. Mit der Eigenschaft **Link zu Objekt** ist es möglich, einen derartigen Verweis im Snapform-Formular abzubilden.

Wenn diese Eigenschaft gewählt wird, öffnet sich ein Drop Down-Menu mit einer Liste aller Elemente der Formularebene (alle Elemente bei welchen die Tabreihenfolge einen Wert grösser als 0 aufweist, also auch Bildfelder). Es kann nun ein Feld gewählt werden. Beim Ausfüllen des Formulars erscheint nun rechts des Feldes eine grüne Pfeilmarkierung (siehe auch Fig. 4-52). Wenn diese Markierung angeklickt wird, wird das verlinkte Feld angesprungen.

Wie in Abschnitt 4.3.6.1 beschrieben, ist es möglich, ein Beschriftungsfeld mit einem Kontrollkästchen zu verknüpfen, womit das Kontrollkästchen auch durch Anklicken des Beschriftungsfelds umgeschaltet werden kann.

4.4.3.2 Optionsfelder

Um Optionsfelder zu erstellen muss ein Weg vorhanden sein, Kontrollkästchen logisch zu gruppieren, damit sie einen Bezug zueinander erhalten. Das Werkzeug zu diesem Zweck ist der Optionsfelder-Container.

Der Optionsfelder-Container ist ein Feld, welches jedes neu erzeugte und darin platzierte Kontrollkästchen zu einem Bestandteil der Optionsfelder-Gruppe macht. In der Objektstruktur sind diese Kontrollkästchen eine Ebene tiefer als die übrigen Felder. Der Optionsfelder-Container gehört eigentlich nicht zur Formelebene, da er nicht Bestandteil der Tabulatoren-Reihenfolge ist (Tab-Reihenfolge Wert -1). Er ist ausserhalb der Entwurfs-Ansicht auch unsichtbar (Vorschau und Snapform Viewer).

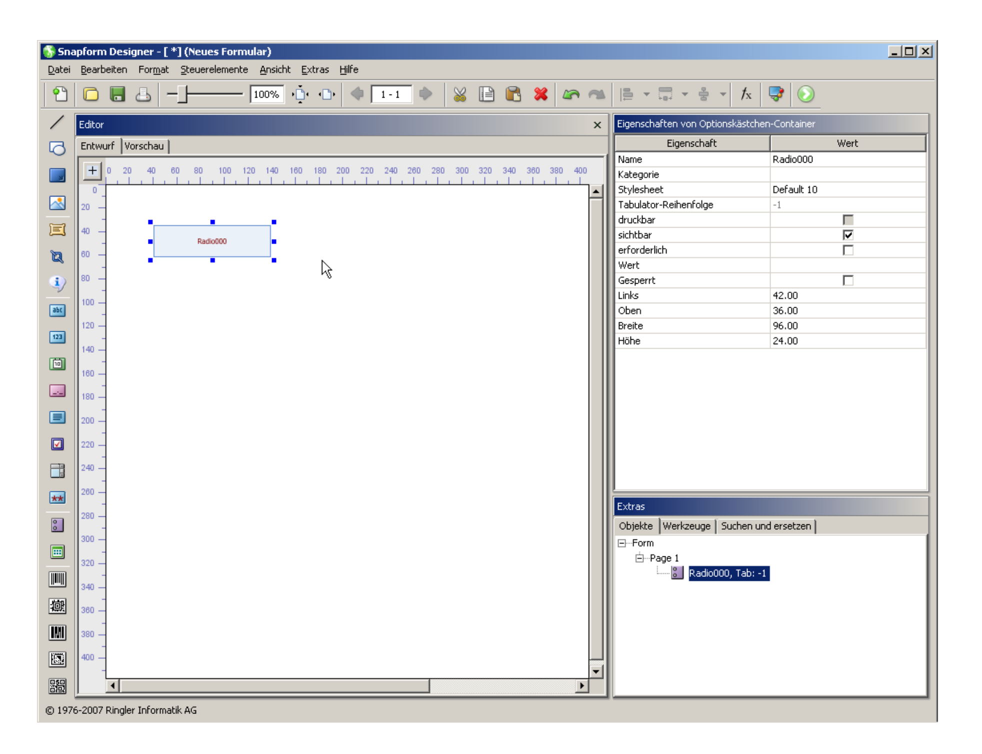
Das Optionsfelder-Werkzeug erzeugt einen Optionsfelder-Container. Nach Wahl des Werkzeugs erscheint ein Cursor mit einem Optionsfelder-Container in Default-Grösse (siehe Fig. 4-87).

Fig. 4-87 *Optionsfelder-Cursor*



Mit dem Optionsfelder-Werkzeug verknüpft ist ein Optionsfelder-Container, 96 pt breit und 24 pt hoch. Mit Mausklick wird dieses Feld auf der Arbeitsfläche platziert. Dabei erhält es automatisch einen Namen, bestehend aus den Zeichen Radio und einer dreistelligen fortlaufenden Nummer, beginnend mit 000. Der erste in einem Formular mit dem Optionsfelder-Werkzeug platzierte Container trägt daher den Namen Radio000. In der Objektstruktur wird er der betreffenden Seite hinzugefügt. In Fig. 4-88 ist der erste Optionsfelder-Container in einem neuen Formular platziert, und anschliessend mit Anklicken selektiert worden.

Fig. 4-88 *Neu erstellter Optionsfelder-Container 060*



Die Länge und Breite des Optionsfelder-Containers kann durch Ziehen der Ankerpunkte des Selektionsrechtecks verändert werden, die Position auch durch Anfassen und Ziehen. Beim Überfahren der Ankerpunkte ändert der Cursor in Doppelpfeile, welche die

Bewegungsrichtung des Ankerpunktes angeben. Beim “Anfassen” und Drücken der Maustaste ändert sich der Cursor zu einem Hohlpfeil.

Die übrigen Merkmale eines Optionsfelder-Containers werden über das Eigenschaften-Fenster kontrolliert (siehe Fig. 4-89).

Fig. 4-89 *Eigenschaften eines Optionsfelder-Containers 060*

Eigenschaften von Optionskästchen-Container

Eigenschaft	Wert
Name	Radio000
Kategorie	
Stylesheet	Default 10
Tabulator-Reihenfolge	-1
druckbar	☐
sichtbar	☑
erforderlich	☐
Wert	
Gesperrt	☐
Links	42.00
Oben	36.00
Breite	96.00
Höhe	24.00

Name

Der Name des Optionsfelder-Containers im Dokument. Dieser Name kann geändert werden.

Kategorie

Zugehörigkeit zu einer oder mehreren Kategorien, welche es erlauben, logisch zusammengehörende Gruppen von Elementen zu bilden. Beim Anklicken dieses Felds öffnet sich der Kategorien Dialog (siehe Fig. 4-23).

Die Kategorien werden durch Anklicken des betreffenden Kontrollkästchens ausgewählt. Mit **OK** wird die Auswahl übernommen.

Stylesheet

Das für das Element gültige Stylesheet. Wenn explizite kein Stylesheet ausgewählt worden ist, wird das erste Default-Stylesheet genutzt.

Tabulator-Reihenfolge

Ordnungsnummer des Elements in der Tabulator-Reihenfolge. Dieser Wert ist immer -1 und kann nicht verändert werden. Das bedeutet, dass das Element nicht Bestandteil der Tabulatoren-Reihenfolge ist.

druckbar

Dieses Kontrollkästchen ist deaktiviert, und kann nicht verändert werden. Optionsfelder-Container werden nie ausgedruckt.

sichtbar

Wenn dieses Kontrollkästchen gewählt ist, wird der Optionsfelder-Container auf dem Bildschirm angezeigt.

erforderlich

Wenn dieses Kontrollkästchen gewählt ist, muss eine Option der Optionsfelder gewählt worden sein, damit das Formular gültig ist (für Formular-Gültigkeit, siehe auch Abschnitt 4.8.4.2).

Wert

Es ist zwar möglich, bei Optionsfelder-Containern einen konstanten Wert zu definieren, aber wenig sinnvoll. Anstelle dessen wird hier die Formel, welche die mit dem Umschalten zwischen den Optionsfeldern verbundenen Aktionen steuert, eingegeben.

Die Formel wird beim Umschalten der Optionsfelder sowie beim Öffnen des Dokuments und im Rahmen der übrigen Formelverarbeitungsläufe ausgeführt.

Gesperrt

Wenn dieses Kontrollkästchen gewählt ist, wird der Optionsfelder-Container einschliesslich der darin enthaltenen Kontrollkästchen-Felder im Snapform Designer blockiert, und kann nicht (versehentlich) verschoben werden. Zudem werden sämtliche übrigen Eigenschaften im Eigenschaften-Dialog deaktiviert (ebenso für die enthaltenen Kontrollkästchen-Felder). Der Zugriff auf die Eigenschaften ist erst wieder möglich, wenn dieses Kontrollkästchen deaktiviert wird.

Links

X-Koordinate der linken oberen Ecke der Umrandungsbox des Elements in Points. Diese Eigenschaft erlaubt eine genaue numerische Positionierung des Optionsfelder-Containers.

Oben

Y-Koordinate der linken oberen Ecke der Umrandungsbox des Elements in Points. Diese Eigenschaft erlaubt eine genaue numerische Positionierung des Optionsfelder-Containers.

Breite

Breite der Umrandungsbox des Optionsfelder-Containers.

Höhe

Höhe der Umrandungsbox des Optionsfelder-Containers.

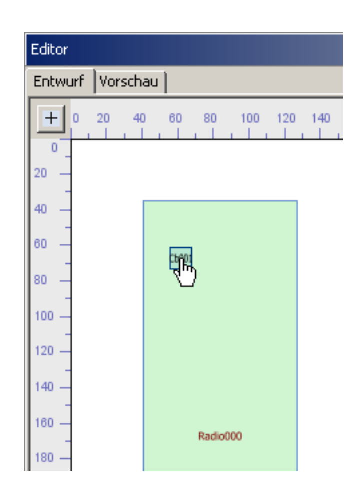
4.4.3.3 Optionsfelder aufbauen

Die Komponenten für Optionsfelder sind in den beiden obigen Abschnitten beschrieben. Es geht nun um das Aufbauen von Optionsfeldern.

Der erste Schritt ist das Bereitstellen des Optionsfelder-Containers. Der Einfachheit halber sollte dieser den Bereich der Optionsfelder abdecken.

Der zweite Schritt ist das Platzieren von Kontrollkästchen innerhalb des Optionsfelder-Containers. Diese Kontrollkästchen werden automatisch dem Optionsfelder-Container zugeordnet, und werden dadurch zu Optionsfeldern. Wenn ein Kontrollkästchen in einen Optionsfelder-Container platziert wird, ändert dieser seine Farbe (siehe Fig. 4-90).

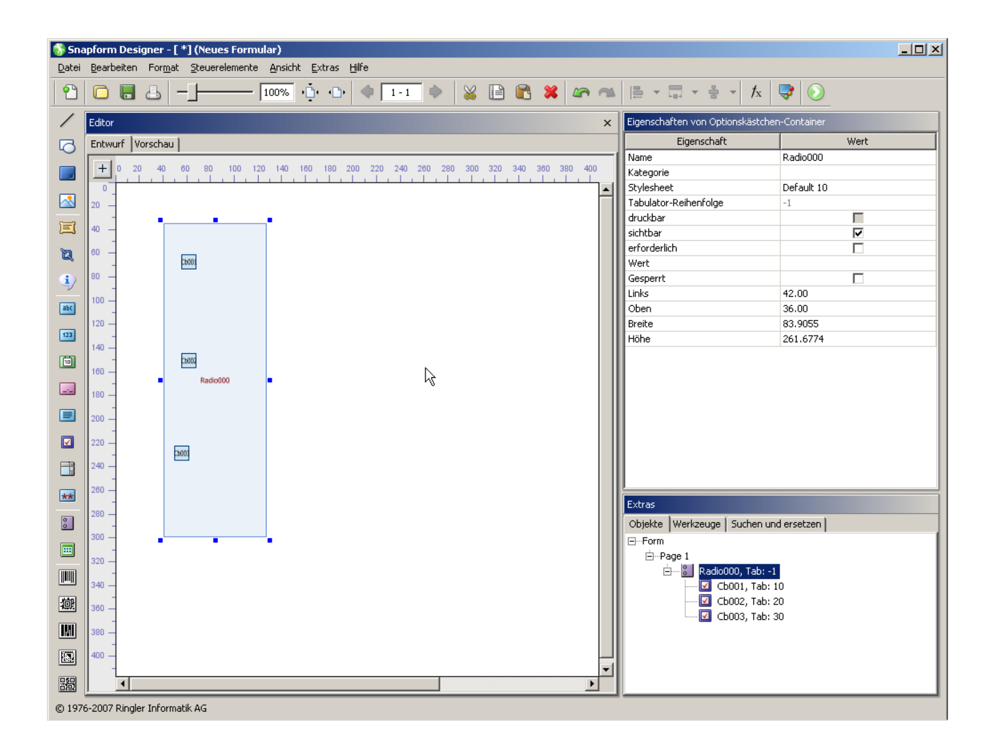
Fig. 4-90 *Kontrollkästchen in Optionsfelder-Container platzieren 061*



Die Farbänderung ist die visuelle Anzeige, dass das Kontrollkästchen angenommen wird.

Dieser Schritt wird nun so lange wiederholt, bis für sämtliche betreffenden Optionen ein Kontrollkästchen vorhanden ist (siehe Fig. 4-91).

Fig. 4-91 *Optionsfelder-Container mit platzierten Kontrollkästchen 062*



Die Dokumentstruktur zeigt in Beispiel die Zugehörigkeit der drei Kontrollkästchen Cb001, Cb002 und Cb003 zum Optionsfelder-Container Radio000.

Kontrollkästchen, welche vor dem Platzieren des Optionsfelder-Containers bereits auf dem Formular vorhanden waren, können nicht zur Gruppe der Optionsfelder hinzugefügt werden, ausser sie werden mit Ausschneiden/Einfügen neu platziert (dabei ist dann auch die Tabulator-Reihenfolge zu kontrollieren). Ebensowenig können später ausserhalb des Bereichs des Optionsfelder-Containers platzierte Kontrollkästchen hinzugefügt werden.

Kontrollkästchen-Felder, welche im Optionsfelder-Container platziert worden sind, können nicht aus dem Container herausgezogen werden. Andererseits ist es möglich, die Grösse des Optionsfelder-Containers zu ändern, so dass zugehörige Kontrollkästchen-Felder ausserhalb zu liegen kommen. Diese Felder sind nach wie vor Bestandteil der Optionsfelder-Gruppe, und sie werden beim Verschieben des Optionsfelder-Containers genau so mit bewegt.

Es kann der Fall auftreten, bei welchen die Option für ein Kontrollkästchen bzw. Optionsfeld wiederholt wird (zum Beispiel in einem Formular mit Rückantwort-Talon). Hierzu wird an der betreffenden Stelle ein Kontrollkästchen eingefügt. Falls es sich um Optionsfelder handelt, muss darauf geachtet werden, dass das neu eingefügte Kontrollkästchen nicht Bestandteil der Optionsfelder-Gruppe wird. Dieses neue Kontrollkästchen wird schreibgeschützt gesetzt (damit es nicht manuell übersteuert werden kann). Angenommen, das Kontrollkästchen solle den Zustand von Cb001 wiederholen, wird nun in Wert die Formel

```
= Cb001
```

eingegeben. Und bei der Kontrolle in der Vorschau gibt dieses Kontrollkästchen immer den Zustand von Cb001 wieder.

4.4.3.4 Beispiele von Kontrollkästchen und Optionsfeldern

Das Dokument checkbuttons.qdf (siehe Fig. 4-92 (Entwurfsmodus)) bzw. Fig. 4-93 (Ansicht in Snapform Viewer, ausgefüllt)) sind verschiedene Beispiele von Kontrollkästchen und Optionsfeldern gezeigt.

Nach dem Selektieren der Feldelemente im Snapform Designer gibt das Eigenschaften-Fenster Auskunft über dessen jeweiligen Merkmale. Ein Doppelklick auf das Feld öffnet den Formeleditor zur Anzeige der Formel (sofern vorhanden).

Fig. 4-92 *Beispiele von checkbuttons.qdf, im Entwurfs-Modus 063*

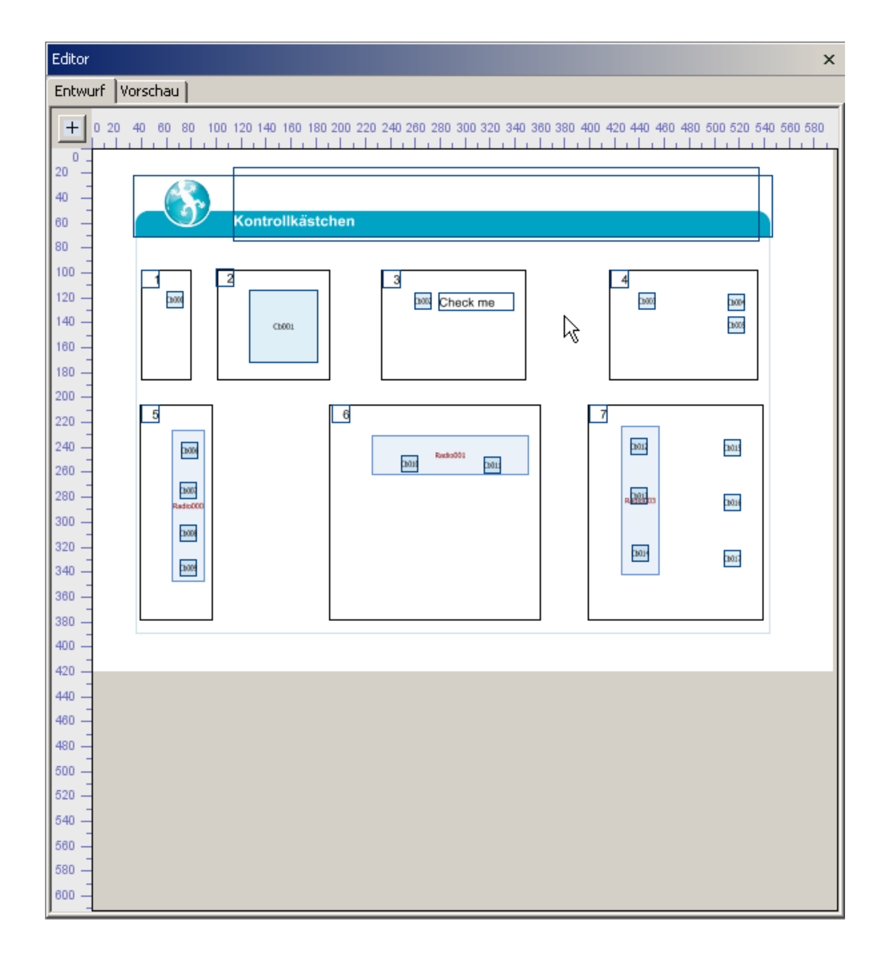
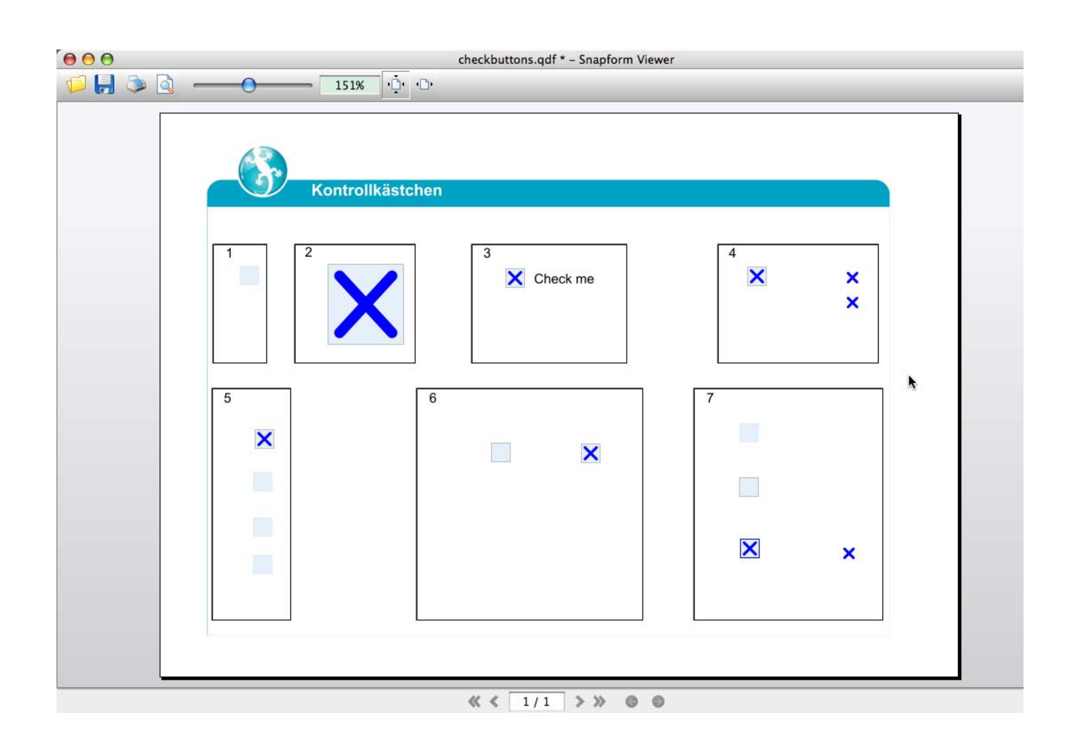


Fig. 4-93 *Ausgefüllte Beispiele von checkbuttons.qdf in Snapform Viewer 064*



1 Cb000: Einfaches Kontrollkästchen in Ursprungsgrösse
2 Cb001: Grosses Kontrollkästchen
3 Cb002: Kontrollkästchen, welches mit einem Beschriftungsfeld Lb000 verknüpft ist.
4 Cb003: Kontrollkästchen mit zwei abhängigen Kontrollkästchen Cb004 und Cb005; letztere sind schreibgeschützt
5 Radio000: Optionsfeld-Container mit vier Kontrollkästchen Cb006, Cb007, Cb008 und Cb009
6 Radio001: Optionsfeld-Container mit zwei Kontrollkästchen Cb010 und Cb011; die Grösse des Containers wurde verändert, so dass die Kontrollkästchen nun ausserhalb liegen
7 Radio002: Optionsfeld-Container mit drei Kontrollkästchen Cb012, Cb013 und Cb014 und jeweils einem abhängigen Kontrollkästchen Cb015, Cb016 und Cb017; letztere sind schreibgeschützt und geben den Zustand der Optionsfelder wieder

4.4.4 Auswahllisten

Ein häufiges Formularelement ist eine Liste von Möglichkeiten, aus welchen ausgewählt werden kann. Dieses Element hat kein Äquivalent in der Papierform, ist jedoch in elektronischen Formularen weit

verbreitet (und in diesem Fall auch einer grösseren Reihe von Optionsfeldern vorzuziehen.

Auswahllisten benutzen zwei Informationen: die Listenwert und den Auswahlwert. Der Listenwert wird beim Aufklappen der Liste angezeigt, und der Auswahlwert erscheint anschliessend im Listenfeld. Dies ist auch der Wert, welcher in Formeln weiter verarbeitet werden kann.

Wenn kein Auswahlwert vorhanden ist, wird der Listenwert benutzt.

Die Auswahllisten können auch über Formeln zusammengesetzt werden.

Auswahllistenfelder bieten eine Auswahl aus einer fest definierten oder dynamisch aufgebauten Liste von Möglichkeiten.

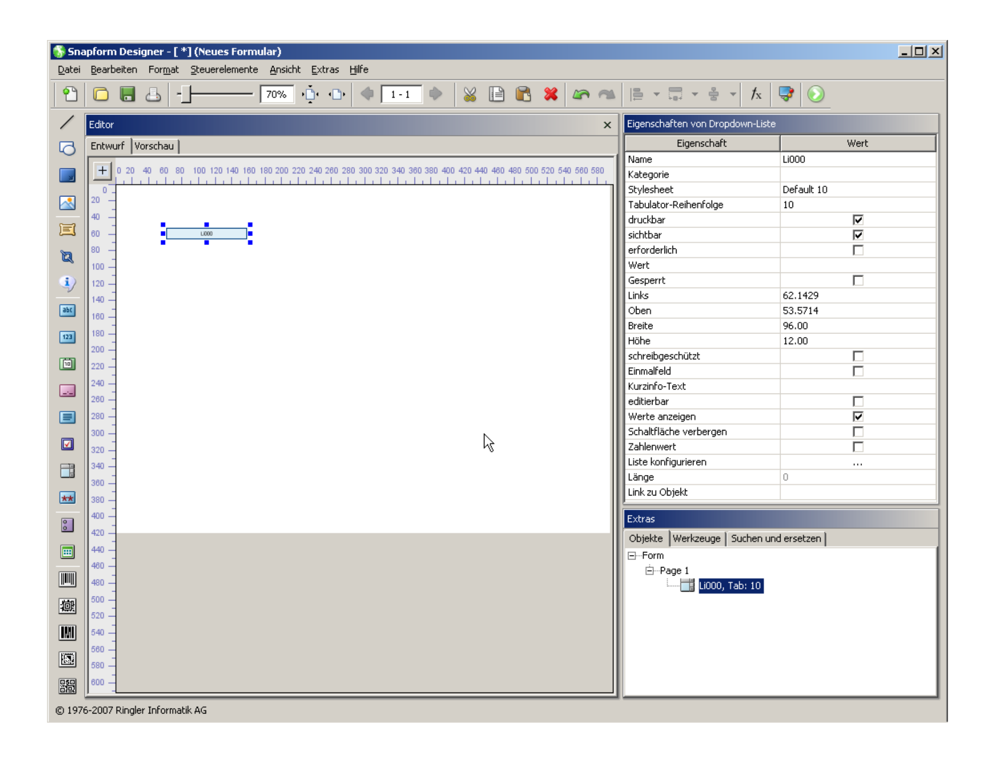
Das Auswahllistenfeldwerkzeug erzeugt ein Feld, welches die Auswahl aus einer Liste von Möglichkeiten erlaubt. Nach Wahl des Werkzeugs erscheint ein Cursor mit einem Auswahllistenfeld in Default-Grösse (siehe Fig. 4-94).

Fig. 4-94 *Auswahllistenfeld-Cursor*



Mit dem Auswahllistenfeld-Werkzeugt verknüpft ist ein Auswahllistenfeld, 96 pt breit und 12 pt hoch. Mit Mausklick wird dieses Feld auf der Arbeitsfläche platziert. Dabei erhält es automatisch einen Namen, bestehend aus den Zeichen Li und einer dreistelligen fortlaufenden Nummer, beginnend mit 000. Das erste in einem Formular mit dem Auswahllistenwerkzeug platzierte Feld trägt daher den Namen Li000. In der Objektstruktur wird es der betreffenden Seite hinzugefügt. In Fig. 4-95 ist das erste Auswahllistenfeld in einem neuen Formulars platziert, und anschliessend mit Anklicken selektiert worden.

Fig. 4-95 *Neu erstelltes Auswahllistenfeld 065*



Die Länge und Breite des Auswahllistenfelds kann durch Ziehen der Ankerpunkte des Selektionsrechtecks verändert werden, die Position auch durch Anfassen und Ziehen. Beim Überfahren der Ankerpunkte ändert der Cursor in Doppelpfeile, welche die Bewegungsrichtung des Ankerpunktes angeben. Beim "Anfassen" und Drücken der Maustaste ändert sich der Cursor zu einem Hohlpfeil.

Die übrigen Merkmale eines Auswahllistenfelds werden über das Eigenschaften-Fenster kontrolliert (siehe Fig. 4-96).

Fig. 4-96 *Eigenschaften eines Auswahllistenfelds 065*

Eigenschaften von Dropdown-Liste

Eigenschaft	Wert
Name	Li000
Kategorie	
Stylesheet	Default 10
Tabulator-Reihenfolge	10
druckbar	☑
sichtbar	☑
erforderlich	☐
Wert	
Gesperrt	☐
Links	62.1429
Oben	53.5714
Breite	96.00
Höhe	12.00
schreibgeschützt	☐
Einmalfeld	☐
Kurzinfo-Text	
editierbar	☐
Werte anzeigen	☑
Schaltfläche verbergen	☐
Zahlenwert	☐
Liste konfigurieren	...
Länge	0
Link zu Objekt	

Name

Der Name des Auswahllistenfelds im Dokument. Dieser Name kann geändert werden.

Kategorie

Zugehörigkeit zu einer oder mehreren Kategorien, welche es erlauben, logisch zusammengehörende Gruppen von Elementen zu bilden. Beim Anklicken dieses Felds öffnet sich der Kategorien Dialog (siehe Fig. 4-23).

Die Kategorien werden durch Anklicken des betreffenden Kontrollkästchens ausgewählt. Mit **OK** wird die Auswahl übernommen.

Stylesheet

Das für das Element gültige Stylesheet. Wenn explizite kein Stylesheet ausgewählt worden ist, wird das erste Default-Stylesheet genutzt. Für Auswahllistenfelder sind vor allem die **Allgemeinen Einstellungen** von Belang.

Tabulator-Reihenfolge

Ordnungsnummer des Elements in der Tabulator-Reihenfolge. Dieser Wert wird automatisch vergeben, und liegt in der Reihe der Feld-Tabulator-Reihenfolge (welche mit 10 beginnend in Zehnerschritten erhöht wird). Das erste Feld des Formulars hat daher den Tabulator-Reihenfolge-Wert 10, das zweite den Wert 20, etc.. Der Wert kann

verändert werden, so dass eine frei bestimmbare Tabulatorreihenfolge aufgebaut werden kann, welche nicht mit der Reihenfolge des Hinzufügens der Felder übereinstimmen muss.

druckbar

Wenn dieses Kontrollkästchen gewählt ist, wird das Auswahllistenfeld ausgedruckt.

sichtbar

Wenn dieses Kontrollkästchen gewählt ist, wird das Auswahllistenfeld auf dem Bildschirm angezeigt.

erforderlich

Wenn dieses Kontrollkästchen gewählt ist, muss eine Auswahl aus der Auswahlliste erfolgt sein, damit das Formular gültig ist (für Formular-Gültigkeit, siehe auch Abschnitt 4.8.4.2).

Wert

Es ist zwar möglich, bei Auswahllistenfeldern einen konstanten Wert zu definieren, aber wenig sinnvoll, da er nicht angezeigt wird. Anstelle dessen wird hier die Formel, welche die beim Vornehmen einer Auswahl verbundene Aktionen steuert, eingegeben.

Die Formel wird beim Wechsel des gewählten Listenelements sowie beim Öffnen des Dokuments und im Rahmen der übrigen Formelverarbeitungsläufe ausgeführt.

Gesperrt

Wenn dieses Kontrollkästchen gewählt ist, wird das Auswahllistenfeld im Snapform Designer blockiert, und kann nicht (versehentlich) verschoben werden. Zudem werden sämtliche übrigen Eigenschaften im Eigenschaften-Dialog deaktiviert. Der Zugriff auf die Eigenschaften ist erst wieder möglich, wenn dieses Kontrollkästchen deaktiviert wird.

Links

X-Koordinate der linken oberen Ecke der Umrandungsbox des Elements in Points. Diese Eigenschaft erlaubt eine genaue numerische Positionierung des Auswahllistenfelds.

Oben

Y-Koordinate der linken oberen Ecke der Umrandungsbox des Elements in Points. Diese Eigenschaft erlaubt eine genaue numerische Positionierung des Auswahllistenfelds.

Breite

Breite der Umrandungsbox des Auswahllistenfelds.

Höhe

Höhe der Umrandungsbox des Auswahllistenfelds.

schreibgeschützt

Wenn dieses Kontrollkästchen gewählt ist, ist das Auswahllistenfeld als schreibgeschützt markiert, und dessen Inhalt kann nicht manuell verändert werden. Der Wert kann nach wie vor über Formeln verändert werden. Ein schreibgeschütztes Auswahllistenfeld ist nach wie vor in der Tab-Reihenfolge eingebunden, und dessen Formeln werden auch entsprechend der Auswahl ausgeführt.

Einmalfeld

Es gibt Informationen, welche nach dem ursprünglichen Ausfüllen des Formulars nicht mehr verändert werden dürfen und daher geschützt werden müssen. In Snapform erfolgt dieser Schutz mit Einmalfeldern, welche beim Ausfüllen bis zum ersten Speichern verändert werden können, anschliessend aber als schreibgeschützt markiert werden.

Wenn dieses Kontrollkästchen gewählt ist, kann das Auswahllistenfeld bis zum ersten Speichern eines ausgefüllten Formulars umgeschaltet werden. Nach dem ersten Speichern des Formulars wird das Auswahllistenfeld als schreibgeschützt markiert.

Kurzinfo-Text

Die Kurzinfo ist ein Hilfetext, welcher erscheint, wenn der Mauscursor während einer gewissen Zeit über dem Feld liegt. Dieser Text ist in vielen Fällen die erste und einfachste Hilfe zum betreffenden Feld. Mit zunehmenden Anforderungen an die Barrierefreiheit von elektronischen Formularen erhält der Kurzinfo-Text eine zunehmend grössere Bedeutung, da Screenreader diese Information benutzen. Es ist daher, besonders für Formulare im Behörden-Umfeld, wichtig, einen gut abgefassten Kurzinfo-Text einzufügen.

Wenn diese Eigenschaft gewählt wird, öffnet sich das Kurzinfo-Anzeigefenster (siehe Fig. 4-48). Wie der Erläuterungstext angibt, können für die Kurzinfos auch HTML-Texte eingegeben werden. Es gelten hierfür die gleichen Bedingungen wie für Beschreibungsfelder. Für weitere Anmerkungen zu Kurztexten siehe deren Beschreibung im Abschnitt 4.4.2.1.

Hinweis: *Kurztexte sollten wirklich kurz sein. Daher sollte auf übermässigen Einsatz von HTML-Formatierungen, insbesonder auch auf das Einbinden von Bildern verzichtet werden.*

editierbar

Wenn dieses Kontrollkästchen gewählt ist, kann zusätzlich zu den in der Auswahlliste angebotenen Auswahl ein beliebiger eigener Wert eingegeben werden. Dieser Wert bleibt solange erhalten, bis eine neue Auswahl getroffen wird. Der eingegebene Wert wird nicht in die Auswahl übernommen, und wenn eine neue Auswahl getroffen worden wird, ist der eingegebene Wert verloren.

Werte anzeigen

Wenn dieses Kontrollkästchen gewählt ist, wird bei der geöffneten Liste neben dem Listenwert auch der Auswahlwert angezeigt (siehe Fig. 4-97 (Kontrollkästchen aus) bzw. Fig. 4-98 (Kontrollkästchen ein)).

Fig. 4-97 *Auswahl mit Werte anzeigen aus*

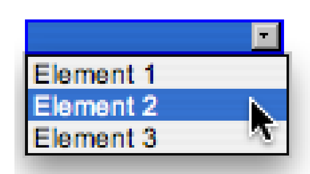
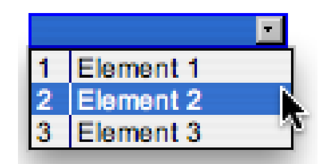


Fig. 4-98 *Auswahl mit Werte anzeigen ein*



Schaltfläche verbergen

Wenn dieses Kontrollkästchen gewählt ist, wird die Schaltfläche am rechten Rand des Auswahllistenfeldes, welche normalerweise immer angezeigt wird (um das Feld als Auswahllistenfeld erkennbar zu machen), ausgeblendet. Diese Schaltfläche ist dann nur sichtbar, wenn das Feld aktiv ist.

Ausrichtung horizontal

Diese Eigenschaft bestimmt, wie der gewählte bzw. eingegebene Text im Auswahllistenfeld ausgerichtet wird.

Wenn der Eigenschaftswert angeklickt wird, erscheint ein Drop Down Menu mit der Auswahl **links**, **zentriert**, **rechts**, welche den Text linksbündig, mittenzentriert bzw. rechtsbündig im Auswahllistenfeld positioniert. Grundeinstellung ist **links**.

Zahlenwert

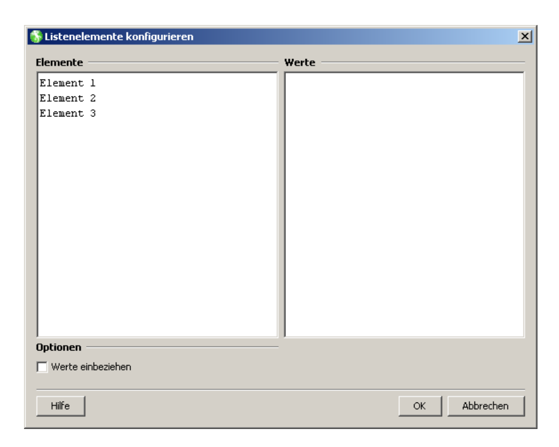
Wenn dieses Kontrollkästchen gewählt ist, muss der Auswahlwert eine Zahl sein.

Liste konfigurieren

Die eigentliche Auswahlliste wird in dieser Eigenschaft festgelegt. Dabei muss der Listenwert immer, der Auswahlwert wenn erwünscht, eingegeben werden.

Wenn für diese Eigenschaft **...** angeklickt wird, öffnet sich das Konfigurationsfenster (siehe Fig. 4-99).

Fig. 4-99 *Konfigurationsfenster für Auswahlliste 066*



Das Konfigurationsfenster besteht aus zwei Textspalten sowie einem Kontrollkästchen im Bereich **Optionen**.

Werte einbeziehen

Wenn dieses Kontrollkästchen gewählt ist, werden die Werte der Spalte **Werte** für die Auswahlliste werwendet. Gleichzeitig wird die Spalte **Werte** auch zugänglich gemacht (wenn das Kontrollkästchen nicht gewählt ist, bleibt die Spalte **Werte** gesperrt).

Elemente

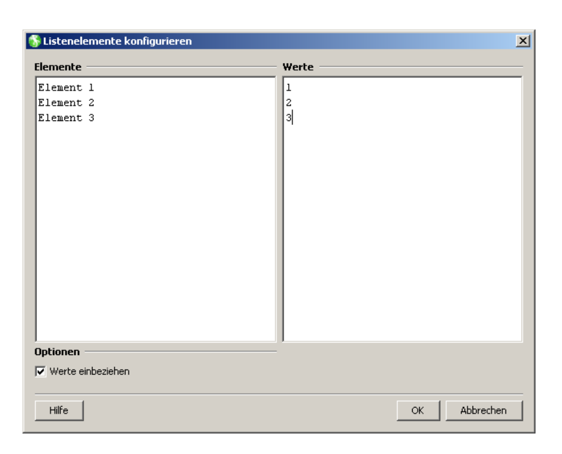
In der Spalte **Elemente** werden die Bezeichnungen der Elemente der Auswahlliste eingegeben. Jedes Element kommt auf eine eigene Zeile. Die Länge der Liste ist nicht begrenzt.

Werte

In der Spalte **Werte** werden die zum jeweiligen Eintrag in **Elemente** zugehörigen Werte eingegeben. Diese Spalte ist nur zugänglich, wenn das Kontrollkästchen Werte einbeziehen gewählt ist (siehe Fig. 4-100).

Hinweis: *Die Spalten Elemente und Werte unterstützen die Eingabe in HTML nicht.*

Fig. 4-100 *Konfigurationsfelder für Auswahlliste mit aktiver Werteliste 067*



Länge

Dieser Wert begrenzt die Länge des manuell eingegebenen Werts auf die betreffende Anzahl Zeichen. Wenn der Wert 0 ist, wird die Länge nicht begrenzt.

Link zu Objekt

In verschiedenen Formularanwendungen gibt es Verweise auf andere Bereiche des Formulars, oder auf ergänzende Informationen. Mit der Eigenschaft **Link zu Objekt** ist es möglich, einen derartigen Verweis im Snapform-Formular abzubilden.

Wenn diese Eigenschaft gewählt wird, öffnet sich ein Drop Down-Menu mit einer Liste aller Elemente der Formularebene (alle Elemente bei welchen die Tabreihenfolge einen Wert grösser als 0 aufweist, also auch Bildfelder). Es kann nun ein Feld gewählt werden. Beim Ausfüllen des Formulars erscheint nun rechts des Feldes eine grüne Pfeilmarkierung (siehe auch Fig. 4-52). Wenn diese Markierung angeklickt wird, wird das verlinkte Feld angesprungen.

Eine Auswahlliste kann auch dynamisch mit Hilfe einer Formel aufgebaut werden. Die hierzu verwendete Funktion heisst `setList()` und ist in Abschnitt 4.5.2.9 beschrieben.

4.4.5 1D-Barcodes

1D-Barcodes erlauben die Codierung von geringen Datenmengen als Strichcode, wodurch auch in Papier-basierten Arbeitsabläufen ein Medienbruch vermieden werden kann. Allerdings ist die codierbare Datenmenge sehr gering (je nach System bis zu 40 Zeichen). Daher sind 1D-Barcodes für die Übertragung von Nutzdaten eher ungeeignet. 1D-Barcodes sind jedoch ideal für die Codierung von Daten über das Dokument (Identifikation, Routing-Information, Geschäftsfall-Identifikation, etc.) beispielsweise für Archivierungs- oder Ablagesysteme.

Die Auswahl des “richtigen” 1D-Barcodes hängt von verschiedenen Aspekten ab (zum Beispiel Zeichenumfang, Datenlängen, Lesesicherheit, Verarbeitungsleistung, gesetzliche Vorgaben oder vorhandene Infrastruktur) und muss im Rahmen der Analyse betrachtet werden. In diesem Abschnitt wird daher auf eine weitergehende Beschreibung der jeweiligen Barcode-Systeme verzichtet. Anstelle dessen sei auf die einschlägige Literatur verwiesen. Eine kurze

Beschreibung der unterstützten Barcodes befindet sich indessen in Abschnitt 4.4.5.2.

4.4.5.1 Das Barcodewerkzeug

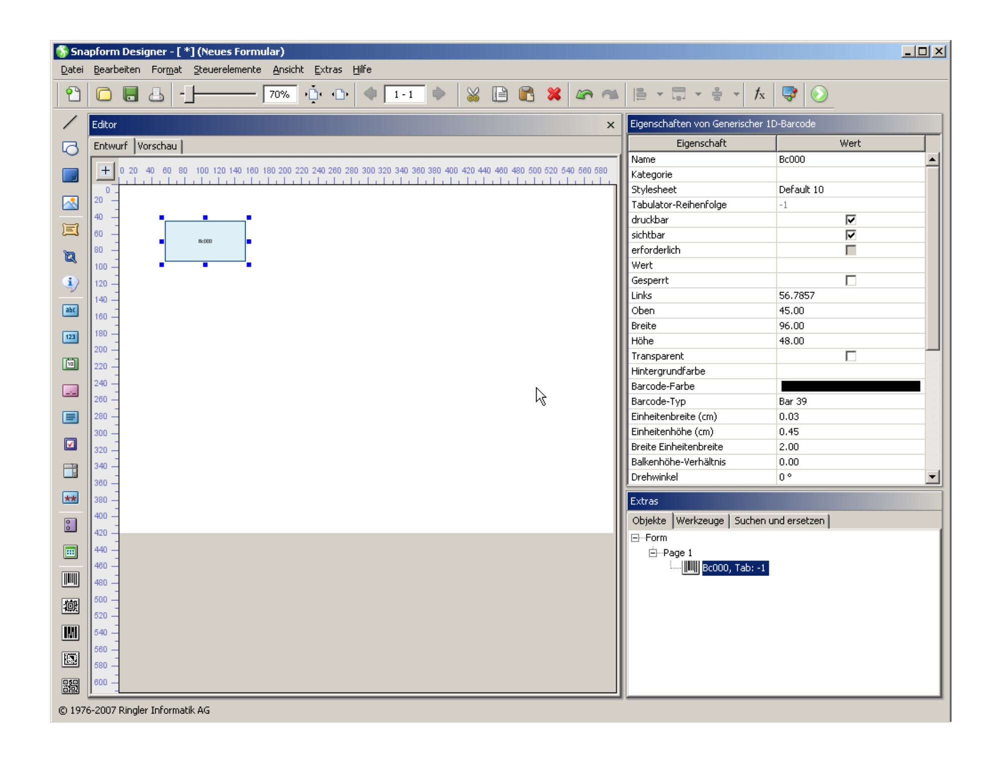
Das Barcodewerkzeug erzeugt ein Feld, in welchem ein 1D-Barcode angezeigt werden kann. Nach Wahl des Werkzeugs erscheint ein Cursor mit einem Barcode-Element in Default-Grösse (siehe Fig. 4-101).

Fig. 4-101 *1D-Barcode-Cursor*



Mit dem Barcodewerkzeug verknüpft ist ein 1D-Barcodefeld, 96 pt breit und 48 pt hoch. Mit Mausklick wird dieses Feld auf der Arbeitsfläche platziert. Dabei erhält es automatisch einen Namen, bestehend aus den Zeichen Bc und einer dreistelligen fortlaufenden Nummer, beginnend mit 000. Das erste in einem Formular mit dem Barcodewerkzeug platzierte Element trägt daher den Namen Bc000. In der Objektstruktur wird es der betreffenden Seite hinzugefügt. In Fig. 4-102 ist das erste Barcodefeld in einem neuen Formular platziert, und anschliessend mit Anklicken selektiert worden.

Fig. 4-102 *Neu erstelltes Barcodefeld 068*



Die Länge und Breite des Barcodefelds kann durch Ziehen der Ankerpunkte des Selektionsrechtecks verändert werden, die Position auch durch Anfassen und Ziehen. Beim Überfahren der Ankerpunkte ändert der Cursor in Doppelpfeile, welche die Bewegungsrichtung des Ankerpunktes angeben. Beim "Anfassen" und Drücken der Maustaste ändert sich der Cursor zu einem Hohlpfeil.

Die übrigen Merkmale einer Barcodefelds werden über das Eigenschaften-Fenster kontrolliert (siehe Fig. 4-103).

Hinweis: *Das Barcodefeld hat verschiedene Eigenschaften, welche nur für bestimmte Barcodesysteme relevant sind. Diese Eigenschaften werden zwar angezeigt, sind aber nur dann aktiv, wenn der zutreffende Barcode gewählt worden ist.*

Fig. 4-103 *Eigenschaften eines Barcodefelds 068/069*

Eigenschaften von Generischer 1D-Barcode

Eigenschaft	Wert
Name	Bc000
Kategorie	
Stylesheet	Default 10
Tabulator-Reihenfolge	-1
druckbar	☑
sichtbar	☑
erforderlich	☐
Wert	
Gesperrt	☐
Links	56.7857
Oben	45.00
Breite	96.00
Höhe	48.00
Transparent	☐
Hintergrundfarbe	
Barcode-Farbe	
Barcode-Typ	Bar 39
Einheitenbreite (cm)	0.03
Einheitenhöhe (cm)	0.45
Breite Einheitenbreite	2.00
Balkenhöhe-Verhältnis	0.00
Drehwinkel	0 °
Text	☑
Codabar-Startzeichen	A
Codabar-Stopzeichen	B
Code128-Zeichensatz	A
Prüfziffer	☐
Schutzbalken	☑
Trennlücke	1.00
PostNet Hohe Balkenhöhe	0.25
PostNet Kleine Balkenhöhe	0.125
Zusatzcode Höhe	0.80
Abstand Zusatzcode (cm)	0.50
2-stelliger Zusatzcode	☐
5-stelliger Zusatzcode	☐
Zusatzcode	

Name

Der Name des Barcodefelds im Dokument. Dieser Name kann geändert werden.

Kategorie

Zugehörigkeit zu einer oder mehreren Kategorien, welche es erlauben, logisch zusammengehörende Gruppen von Elementen zu bilden. Beim Anklicken dieses Felds öffnet sich der Kategorien Dialog (siehe Fig. 4-23).

Die Kategorien werden durch Anklicken des betreffenden Kontrollkästchens ausgewählt. Mit **OK** wird die Auswahl übernommen.

Stylesheet

Das für das Element gültige Stylesheet. Wenn explizite kein Stylesheet ausgewählt worden ist, wird das erste Default-Stylesheet genutzt. Bei Barcodefeldern wird von den Stylesheet-Definitionen einzig Randeinstellungen in Allgemeine Einstellungen berücksichtigt.

Tabulator-Reihenfolge

Ordnungsnummer des Elements in der Tabulator-Reihenfolge. Dieser Wert ist immer -1 und kann nicht verändert werden. Das bedeutet, dass das Element nicht Bestandteil der Tabulatoren-Reihenfolge ist, was an sich logisch ist, da ein Barcode nicht ein interaktives Element ist.

druckbar

Wenn das Kontrollkästchen gewählt ist, erscheint das Barcodefeld im Ausdruck.

sichtbar

Wenn das Kontrollkästchen gewählt ist, erscheint das Barcodefeld auf dem Bildschirm.

erforderlich

Diese Eigenschaft ist bei Barcodefeldern immer ausgeschaltet, und kann nicht geändert werden.

Wert

Mit Anklicken dieser Eigenschaft öffnet sich der Formeleditor zur Eingabe einer Formel.

Barcodefelder können entweder einen konstanten Wert enthalten (zum Beispiel für die Kennung eines Dokuments) oder das Resultat einer Formel (wesentlich häufiger).

Hinweis: *Allfällige Prüfziffern des Barcodes werden automatisch berechnet, und dürfen daher nicht in der Formel berücksichtigt werden.*

Dasselbe gilt auch für die Start- und Stop-Zeichen bei Codabar, welche über die betreffende Barcode-Eigenschaft definiert werden.

Gesperrt

Wenn das Kontrollkästchen gewählt ist, wird das Barcodefeld im Snapform Designer blockiert, und kann nicht (versehentlich) verschoben werden. Zudem werden sämtliche übrigen Eigenschaften im Eigenschaften-Dialog deaktiviert. Der Zugriff auf die Eigenschaften ist erst wieder möglich, wenn dieses Kontrollkästchen deaktiviert wird.

Links

X-Koordinate der linken oberen Ecke der Umrandungsbox des Elements in Points. Diese Eigenschaft erlaubt eine genaue numerische Positionierung des Barcodefelds.

Oben

Y-Koordinate der linken oberen Ecke der Umrandungsbox des Elements in Points. Diese Eigenschaft erlaubt eine genaue numerische Positionierung des Barcodefelds.

Breite

Breite des Barcodefelds in Points.

Hinweis: *Die effektive Breite des Barcodes richtet sich nach der effektiven Anzahl Zeichen und kann wesentlich breiter sein als das Feld. Der Code wird beginnend von der linken oberen Ecke des Felds nach rechts erzeugt. Es muss daher darauf geachtet werden, dass ausreichend Platz vorhanden ist.*

Höhe

Höhe des Barcodefelds in Points.

Transparent

Der Hintergrund des Barcodefelds wird mit diesem Kontrollkästchen auf transparent (keine deckende Hintergrundsfarbe) geschaltet. Diese Eigenschaft ist in der Grundeinstellung ausgeschaltet.

Hintergrundfarbe

Farbe des Barcodefeldes. Mit Anklicken der Eigenschaft öffnet sich das Farbauswahlfenster (siehe Abschnitt 3.2.9.2), welches die Definition der gewünschten Farbe erlaubt.

Die Hintergrundfarbe wird nur dann berücksichtigt, wenn das Kontrollkästchen **Transparent** ausgeschaltet ist.

Barcodefarbe

Farbe des Barcodes. Mit Anklicken der Eigenschaft öffnet sich das Farbauswahlfenster (siehe Abschnitt 3.2.9.2), welches die Definition der gewünschten Farbe erlaubt.

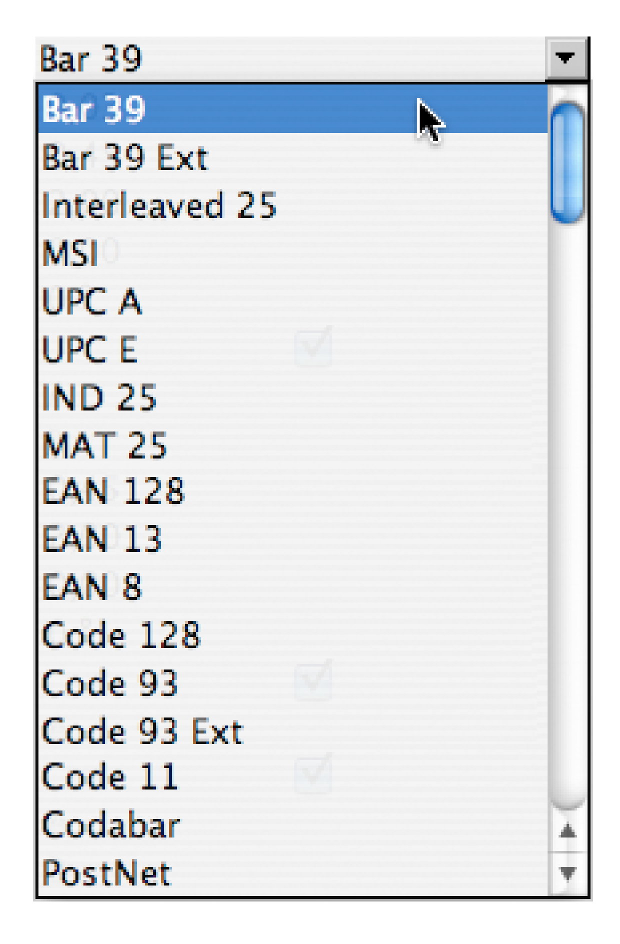
Hinweis: *Die Normen für die betreffenden Barcodes können Vorschriften bezüglich der zu verwendenden Farben enthalten, und sollten konsultiert werden. Ebenso ist zu beachten, dass einzig die Vollfarben Schwarz, Cyan,*

Magenta und Yellow vollflächig gedruckt werden (es sei denn, es würden Schmuckfarben eingesetzt), und dass eine gute Druckqualität nur mit diesen Farben bei 100% Sättigung zu nicht aufgerasterten Balken führt.

Barcodetyp

Typ des anzuzeigenden Barcodes. Mit Anklicken der Eigenschaft öffnet sich ein Drop Down Menu, mit der Liste der verfügbaren Barcodes (siehe Fig. 4-104).

Fig. 4-104 *Liste der verfügbaren 1D-Barcodes*



In Abhängigkeit der getroffenen Auswahl werden Eigenschaften aktiviert oder deaktiviert (siehe Beschreibung der jeweiligen Eigenschaften).

Die folgenden Barcodes werden unterstützt (für eine etwas genauere Beschreibung siehe Abschnitt 4.4.5.2):

Bar 39: Code 39

Bar 39 Ext: Code 39 Extended

Interleaved 25: Interleaved Code 2 of 5

MSI

UPC A

UPC E

IND 25: Industrial Code 2 of 5, Standard Code 2 of 5

MAT 25: MAT25

EAN 128

EAN 13

EAN 8

Code 128

Code 93

Code 93 Ext: Code 93 Extended

Code 11

Codabar

PostNet

Einheitenbreite (cm)

Breite des Einheitsbalkens des Barcodes in Zentimetern. Dieses Mass ist mitbestimmend für die Länge des resultierenden Codes. Bei der Wahl dieses Werts die Auflösung der zu erwartenden Drucker berücksichtigt werden.

Praxistip: *Zum Errechnen der ungefähren Grösse in Points den Wert mit 200 multiplizieren und 10% abziehen.*

Einheitenhöhe (cm)

Höhe des Einheitsbalkens des Barcodes in Zentimetern. Dieses Mass bestimmt direkt die Höhe des resultierenden Codes.

Praxistip: *Zum Errechnen der ungefähren Grösse in Points den Wert mit 200 multiplizieren und 10% abziehen.*

Breite Einheitenbreite

1D-Barcodes haben schmale und breite Einheitsbalken. Dieser Eigenschaftswert ist das Breitenverhältnis zwischen dem breiten und dem schmalen Einheitsbalken. Dieser Wert ist mitbestimmend für die Länge des resultierenden Codes.

Typische Werte sind 2 und 3, wobei für die meisten Barcodesysteme 3 vorgeschrieben oder zumindest vorzuziehen ist, insbesondere bei geringer Einheitenbreite. Dadurch wird die Lesesicherheit des Codes verbessert.

Balkenhöhe-Verhältnis

//Unbekannte Eigenschaft

Drehwinkel

Der Barcode kann um Vielfache von 90° gedreht werden. Beim Anklicken dieser Eigenschaft öffnet sich ein Drop Down Menu mit der Auswahl der Drehwinkel **0°**, **90°**, **180°**, **270°**. Grundeinstellung ist 0°.

Hinweis: *Die Richtung, in welche der Barcode wächst entspricht ebenfalls dem Drehwinkel.*

Text

Wenn dieses Kontrollkästchen gewählt ist, wird der Inhalt des Barcodes auch in Klartext angezeigt. Selbstkontrollziffern werden angezeigt, hingegen werden Hilfszeichen, wie Start- und Stop-Zeichen nicht ausgegeben. Es werden alle daten-relevanten Zeichen angezeigt, welche auch bei einer manuellen Erfassung des Strichcodewerts eingegeben werden müssen.

Codabar-Startzeichen und Codabar Stopzeichen

Der Codabar-Code benutzt teilweise nutzbare Start- und Stop-Zeichen, welche nur am Anfang oder Ende des Codes vorkommen dürfen (siehe Beschreibung in Abschnitt 4.4.5.2). Diese Zeichen werden als A, B, C und D bezeichnet. Start- und Stop-Zeichen können gleich oder ungleich sein.

Diese beiden Eigenschaften sind nur bei gewähltem **Codabar** Codetyp aktiv. Wenn eine Eigenschaft gewählt wird, öffnet sich ein Drop Down Menu mit der Auswahl **A**, **B**, **C**, **D**. Grundwert ist Startzeichen **A** und Stopzeichen **B**.

Code 128-Zeichensatz

Der Code 128-Code (sowie der technisch identische EAN-128-Code) benutzt drei unterschiedliche Zeichensätze, A, B und C (siehe Beschreibung in Abschnitt 4.4.5.2).

Mit dieser Eigenschaft, wird der am Anfang des Codes gültige Zeichensatz gewählt. Ein allfälliger Zeichensatzwechsel innerhalb des Codes wird automatisch markiert.

Diese Eigenschaft ist nur bei gewähltem **Code 128** bzw. **EAN-128** Codetyp aktiv. Wenn sie gewählt wird, öffnet sich ein Drop Down Menu mit der Auswahl **A**, **B**, **C**. Grundwert ist Zeichensatz **A**.

Prüfziffer

In verschiedenen Barcode-Systemen kann zusätzlich eine Prüfziffer eingefügt werden, welche die Lesesicherheit erhöht.

Wenn das Kontrollkästchen gewählt ist, wird eine Prüfziffer eingefügt, welche auch im Klartext angezeigt wird. Die Berechnung der Prüfziffer erfolgt entsprechend dem gewählten Barcodetyp.

Hinweis: *Bei den Handelscodes (EAN-13, EAN-8, UPC-A, UPC-E) ist die Prüfziffer Bestandteil der Nutzdaten und diese Eigenschaft hat nur dann einen Einfluss, wenn der Wert des Codes um eine Stelle kürzer ist (12 Stellen für EAN-13, 11 Stellen für UPC-A und UPC-E, 7 Stellen für EAN-8).*

Beim Codabar-Code gibt es keine Prüfziffer.

Beim PostNet-Code, beim Code 128 bzw. EAN-128 sowie beim Code 11 ist die Prüfziffer fester Bestandteil des Codes. In diesen Fällen erzeugt diese Eigenschaft eine zusätzliche Prüfziffer, welche auf Applikationsebene ausgewertet werden könnte. Die eigentliche Prüfziffer des Barcodes wird über die Gesamtheit der Daten berechnet.

Schutzbalken

Bei den Detailhandels-Codes (EAN-13, EAN-8, UPC-A, UPC-E) kann die Klartextzeile zu Beginn und Ende sowie zwischen dem Standard-Herstellercode und dem Produktcode mit verlängerten Balken umrahmt, "geschützt", werden.

Wenn das Kontrollkästchen gewählt ist, wird das Symbol mit verlängerten Balken angezeigt.

Diese Eigenschaft ist nur bei gewähltem **EAN-13**, **EAN-8**, **UPC-A**, oder **UPC-E** Codetyp verfügbar. In den anderen Fällen wird sie ignoriert. Bei Detailhandels-Anwendungen sollte sie immer gewählt sein.

Trennlücke

Verschiedene Barcodes werden als "diskrete Codes" klassifiziert. Bei diesen Codes können einzelne Zeichen-Äquivalente eigenständig vorhanden sein, ohne Bestandteil eines grösseren Codes sein zu müssen. Bei diesen Codes kann ein leerer Zwischenraum zwischen den einzelnen "Zeichen" definiert werden, dessen einzige Aufgabe es ist, einen Abstand zwischen den Zeichen zu halten. Dieser Abstandhalter ist die Trennlücke, und wird als Vielfaches der Einheitenbreite angegeben.

Diese Eigenschaft ist nur bei gewähltem **Code 11**, **Codabar**, **Bar 39** sowie **Bar 39 Ext** Codetyp verfügbar. In den anderen Fällen wird sie ignoriert.

PostNet Hohe Balkenhöhe und PostNet Kleine Balkenhöhe

Beim PostNet-Code erfolgt die Codierung durch unterschiedliche Balkenhöhen. In diesen Eigenschaften werden die absoluten Abmessungen der Balken in Centimetern definiert.

Das Verhältnis zwischen grosser und kleiner Balkenhöhe sowie die absoluten Masse sind in den Normen spezifiziert.

Die Grundwerte 0,125 bzw. 0,25 cm entsprechen den Normen.

Diese Eigenschaften sind nur bei gewähltem **PostNet** Codetyp verfügbar. In allen anderen Fällen werden sie ignoriert.

Relative Höhe der Zusatzcode-Balken

Bei den Detailhandels-Codes (EAN-13, EAN-8, UPC-A, UPC-E) können für Druckerzeugnisse (Bücher, Periodika) zwei- oder 5-stellige Zusatzcodes hinzugefügt werden. Diese Zusatzcodes sind meist weniger hoch als der Hauptcode, und die Klartextzeile befindet sich oberhalb des Codes.

Diese Eigenschaft legt die relative Höhe des Zusatzcodes fest. Ein typischer Wert ist 0.8.

Diese Eigenschaft ist nur bei gewähltem **EAN-13**, **EAN-8**, **UPC-A**, oder **UPC-E** Codetyp verfügbar. In den anderen Fällen wird sie ignoriert.

Abstand zwischen dem Code und dem Zusatzcode (cm)

Bei den Detailhandels-Codes (EAN-13, EAN-8, UPC-A, UPC-E) können für Druckerzeugnisse (Bücher, Periodika) zwei- oder 5-stellige Zusatzcodes hinzugefügt werden. Diese Zusatzcodes sind meist weniger hoch als der Hauptcode, und die Klartextzeile befindet sich oberhalb des Codes.

Diese Eigenschaft legt den Abstand des Zusatzcodes vom Hauptcode fest. Die Masseinheit ist Centimeter.

Diese Eigenschaft ist nur bei gewähltem **EAN-13**, **EAN-8**, **UPC-A**, oder **UPC-E** Codetyp verfügbar. In den anderen Fällen wird sie ignoriert.

Zusatzcode

Bei den Detailhandels-Codes (EAN-13, EAN-8, UPC-A, UPC-E) können für Druckerzeugnisse (Bücher, Periodika) zwei- oder 5-stellige Zusatzcodes hinzugefügt werden. Diese Zusatzcodes sind meist weniger hoch als der Hauptcode, und die Klartextzeile befindet sich oberhalb des Codes.

Diese Eigenschaft bestimmt den als Zusatzcode anzuzeigenden Text fest. Damit der Zusatzcode angezeigt wird, muss zudem die betreffende Option (**2-stelliger Zusatzcode** bzw. **5-stelliger Zusatzcode**) gewählt sein. Der Code wird zudem nur angezeigt, wenn der Wert aus 2 bzw. 5 Ziffern besteht.

Diese Eigenschaft ist nur bei gewähltem **EAN-13**, **EAN-8**, **UPC-A**, oder **UPC-E** Codetyp verfügbar. In den anderen Fällen wird sie ignoriert.

2-stelliger Zusatzcode und 5-stelliger Zusatzcode

Bei den Detailhandels-Codes (EAN-13, EAN-8, UPC-A, UPC-E) können für Druckerzeugnisse (Bücher, Periodika) zwei- oder 5-stellige Zusatzcodes hinzugefügt werden. Diese Zusatzcodes sind meist weniger hoch als der Hauptcode, und die Klartextzeile befindet sich oberhalb des Codes.

Wenn eines dieser Kontrollkästchen gewählt wird, wird dem Hauptcode ein 2- bzw. 5-stelliger Zusatzcode hinzugefügt. Der Wert des Zusatzcodes ist in der Eigenschaft **Zusatzcode** festgelegt.

Diese Eigenschaft ist nur bei gewähltem **EAN-13**, **EAN-8**, **UPC-A**, oder **UPC-E** Codetyp verfügbar. In den anderen Fällen wird sie ignoriert.

4.4.5.2 Unterstützte 1D-Barcodes

Snapform unterstützt die nachstehend aufgeführten 1D-Barcodes. Diese Beschreibung soll eine kurze Hilfe bei der Auswahl der Codes geben; die Konsultation der Normen bzw. technischen Beschreibungen ist jedoch in den meisten Fällen notwendig.

Bar 39

Dies ist der Code 39, auch bekannt als 3 of 9 oder USD-3. Ein Zeichen wird durch 9 Balken, wovon 3 breit, dargestellt. Dieser Code gilt als der älteste alphanumerische 1D-Barcode. Code 39 ist eine diskrete Symbologie mit variabler Länge. Er ist insofern selbstkontrollierend, dass ein einfacher Druck-Fehler ein Symbol nicht in ein anderes gültiges Symbol überführt werden kann.

Gültige Zeichen: `0123456789[Space]ABCDEFGHIJKLMNOPQRSTUVWXYZ-.$/+%`

Einsatzbereich: Industrie, Medizin, Militär

Eine Prüfziffer modulo 43 kann optional spezifiziert werden.

Bar 39 Ext

Dies ist der Code 39 extended, oder Full-ASCII Code 39. Er basiert auf dem Code 39, erlaubt aber die Darstellung aller 7-bit ASCII-Zeichen, indem Zeichen des Code 39-Zeichenumfangs kombiniert werden.

Gültige Zeichen: 0123456789[Space]ABCDEFGHIJKLMNOPQRSTUVWXYZ!#$%&'()*+,-./:;<=>?@[\]^_` abcdefghijklmnopqrstuvwxyz{|} plus nicht darstellbare Sonderzeichen

Einsatzbereich: Industrie, Medizin, Militär

Eine Prüfziffer Modulo 43 kann optional spezifiziert werden.

IND 25

Dies ist der Industrial 2 of 5 oder Standard 2 of 5 Code, ein numerischer Code niedriger Dichte. Ein Zeichen wird durch 5 schwarze Balken, wovon 2 breit, dargestellt. Die Balkenzwischenräume sind ohne Informationsgehalt und dienen einzig zum Auseinanderhalten der schwarzen Balken. Dieser Code ist einer der ältesten numerischen Codes. Der 2 of 5-Code ist eine diskrete Symbologie mit variabler Länge. Er besitzt keine Selbstkontrollmechanismen. Eine Prüfziffer Modulo 10 kann jedoch auf Applikationsebene definiert werden.

Gültige Zeichen: 0123456789

Einsatzbereich: Industrie

Dieser Code sollte wegen seiner geringen Informationsdichte nicht mehr für neue Anwendungen eingesetzt werden. Anstelle dessen sollte — sofern überhaupt ein 2 of 5-Typ eingesetzt werden soll — der Interleaved 2 of 5-Code zum Zuge kommen.

Interleaved 25

Dies ist der Interleaved 2 of 5 Code, ein numerischer Code höherer Dichte. Er baut auf dem Standard 2 of 5 auf, nutzt jedoch auch die Balkenzwischenräume. Ein Zeichen wird durch 5 schwarze Balken oder weisse Zwischenräume, wovon jeweils 2 breit, dargestellt. Der 2 of 5-Code ist eine diskrete Symbologie mit variabler Länge. Es muss immer eine gerade Anzahl Stellen definiert sein; bei einer ungeraden anzahl

Stellen ist eine 0 voranzustellen. Er besitzt keine Selbstkontrollmechanismen. Eine Prüfziffer Modulo 10 kann jedoch auf Applikationsebene definiert werden.

Gültige Zeichen: `0123456789`

Einsatzbereich: Logistik

Dieser Code ist um etwa ein Drittel kürzer als ein entsprechender Standard 2 of 5-Code.

MAT 25

Dies ist ein sehr selten benutzter Code, für welchen keine weiteren Beschreibungen gefunden werden konnten.

MSI

Dies ist der MSI oder Modified Plessey Code, ein numerischer Code geringer Dichte. Ein Zeichen wird durch eine Folge von Balken oder Zwischenräumen, entsprechend dessen binärem Wert dargestellt. Der MSI-Code ist eine Fortlaufende Symbologie mit variabler Länge. Er besitzt verschiedene Möglichkeiten zur Selbstkontrolle:

- einfache Prüfziffer Modulo 10
- Prüfziffer Modulo 10 über den bereits mit einer Prüfziffer Modulo 10 versehenen Code
- Kombination von Prüfziffer Modulo 11 und Prüfziffer Modulo 10.

Gültige Zeichen: `0123456789`

Einsatzbereich: Bibliotheken, Detailhandel-Logistik

Dieser Code gilt allgemein als veraltet, und wird von modernen Lesegeräten nicht mehr unterstützt.

EAN 13 und UPC-A

Diese beiden Codes sind die sogenannt “grossen” Detailhandelscodes, da sie im Detailhandel weltweit eine grosse Bedeutung haben. Sie sind numerische Codes mit vordefinierter Länge (UPC-A 12 Zeichen, EAN 13 mit 13 Zeichen). Die Struktur ist identisch: 1 (UPC-A) oder 2 (EAN-13)

Zeichen Länder-/Typcode, gefolgt von 5 Zeichen Herstellercode, dann 5 Zeichen Produktcode und 1 Zeichen Prüfziffer. In gewissen Bereichen kann der Ländercode auch 3-stellig sein; dadurch wird der Herstellercode 4-stellig. Bei der EAN-13-Darstellung eines UPC-A-Codes wird eine 0 vorangestellt. Bedingt durch die Codierung ergeben sich identische Symbole, so dass ein mit UPC-A ausgezeichnetes Produkt auch mit einem auf EAN-13 ausgelegten Lesesystem korrekt erfasst wird.

Gültige Zeichen: **`0123456789`**

Einsatzbereich: Detailhandel

Der EAN-13 Code wird, mit einem spezifischen "Ländercode" versehen, auch zur Darstellung von ISBN- und ISSN-Nummern verwendet.

Bei Büchern und Periodika kann der EAN-13 bzw. UPC-A Code mit einem zwei- oder 5-stelligen Zusatzcode versehen werden. Ersterer dient bei Zeitschriften zur Kennzeichnung der Ausgabe (Monat, Woche, etc.) und letzterer dient zur Angabe des empfohlenen Mindestverkaufspreises. Diese Zusatzcodes sind ebenfalls numerisch.

EAN 8 und UPC-E

Diese beiden Codes sind die sogenannten "kleinen" Detailhandelscodes. Bei kleinen Packungen erwies es sich als schwierig, einen vollständigen UPC-A bzw. EAN-13 Code unterzubringen. Aus diesem Grund wurde ein gleich aufgebauter, kleinerer Code entwickelt. Der UPC-E Code ist eine auf 6 Stellen konzentrierte Version der 10 Stellen eines UPC-A Codes.

Hinweis: *Da der UPC-E Code eine "Reduktion" des UPC-A Codes ist, muss als Wert eine gültige und reduktionsfähige 12-stellige UPC-A-Nummer eingegeben werden.*

Der EAN-8 Code ist eine eigenständige Nummer, welche zentral verwaltet wird.

Gültige Zeichen: **`0123456789`**

Einsatzbereich: Detailhandel

Analog zum EAN-13 bzw. UPC-A Code kann der EAN-8 bzw. UPC-E Code mit einem zwei- oder fünfstelligen Zusatzcode ergänzt werden (siehe oben).

Code 128 und EAN-128

Diese beiden Codes sind technisch gesehen identisch. Der Code 128 ist ein alphanumerischer Code hoher Dichte. Code 128 ist eine Fortlaufende Symbologie mit variabler Länge. Der Code enthält eine Prüfsummen Modulo 103, welche automatisch berechnet, und nicht im Klartext angezeigt wird.

Der EAN 128 (auch UCC 128) ist technisch ein Code 128, umfasst jedoch logische Blöcke, welche für die Logistik wichtige Informationen enthalten (z.B. Gewicht und Abmessungen des Behälters). Diese Blöcke beginnen mit einem sogenannten Application Identifier, welcher dann von einer spezifischen Anzahl Zeichen gefolgt wird. Die Application Identifier sind in den Normen spezifiziert.

Er kann sämtliche 127 7-bit ASCII-Zeichen darstellen. Dabei stehen drei verschiedene Zeichensätze zur Verfügung, zwischen welchen innerhalb des Symbols über Steuerzeichen gewechselt werden kann. Die Zeichensätze umfassen folgende Zeichen:

- Zeichensatz A: Ziffern, Grossbuchstaben und Steuer- und Sonderzeichen
- Zeichensatz B: Ziffern, Gross- und Kleinbuchstaben
- Zeichensatz C: Doppelziffern

Gültige Zeichen: **0123456789[Space]ABCDEFGHIJKLMNOPQRSTUVWXYZ!#$%&'()*+,-./:;<=>?@[\]^_` abcdefghijklmnopqrstuvwxyz{|}** plus nicht darstellbare Sonderzeichen

Einsatzbereich: Logistik (insbesondere als EAN 128) und viele andere Anwendungen.

Code 93 und Code 93 Ext

Dies ist eine Erweiterung und Ergänzung zum Code 39, welche es erlaubt, den ganzen 7-bit ASCII-Zeichensatz darzustellen (mit Code 93 extended). Diese Codes sind, im Gegensatz zu Code 39 fortlaufend, was zu einer etwas höheren Dichte führt. Der Code enthält nicht im Klartext angezeigte Prüfsummen.

Gültige Zeichen: **0123456789[Space]ABCDEFGHIJKLMNOPQRSTUVWXYZ-.$/+%**

Zusätzlich in Code 93 Ext: **!#&'()*,:;<=>?@[\]^_` abcdefghijklmnopqrstuvwxyz{|}~** plus nicht darstellbare Sonderzeichen

Anwendung: Industrie

Code 11

Dies ist ein numerischer Code höherer Dichte. Code 11 ist eine diskrete Symbologie mit variabler Länge. Der Code ist nicht sehr sicher, das Fehler im Druck gültige Zeichen erzeugen können. Er besitzt aber eine (bis zu 10 Zeichen Nutzdaten) oder zwei (bei mehr als 10 Zeichen Nutzdaten) Prüfsummen, welche im Klartext nicht angegeben werden.

Gültige Zeichen: **0123456789-**

Einsatzbereich: Telekommunikation in Nordamerika

Codabar

Dies ist ein quasi-numerischer Code mittlerer Dichte. Codabar ist eine diskrete, selbstkontrollierende Symbologie variabler Länge. Der Code beginnt immer mit einem Startzeichen (A, B, C, D), und endet mit einem Stopzeichen (ebenfalls A, B, C, D). Start- und Stopzeichen können auch als Informationsträger dienen. Die einzelnen Zeichen beginnen und enden mit einem Balken. Daher müssen sie mit einem schmalen Zwischenraum getrennt werden.

Gültige Zeichen: **0123456789-$:/.+** plus Start- und Stop-Zeichen

Einsatzbereich: Photolabors, Logistik

Postcode

Dies ist der vom US-Postamt entwickelte PostNet Code, welcher zur automatisierten Sortierung von Postsendungen eingesetzt wird. PostNet ist eine numerische diskrete Symbologie variabler Länge (in der ursprünglichen Anwendung ist die Länge auf 5, 9 oder 11 Zeichen Nutzdaten fixiert). Beim PostNet Code werden die Daten über die Höhe der Balken codiert (was eine sehr schlechte Druckqualität (Matrixdrucker) zulässt). Der Code enthält eine Prüfsumme Modulo 10, welche nicht im Klartext angezeigt wird. Für die Dimensionen und Positionierung des PostNet Codes sind die betreffenden Vorgaben des USPS zu beachten.

Gültige Zeichen: **`0123456789`**

Einsatzbereich: Postsortierung in USA

4.4.5.3 1D-Barcode-Beispiele

Die nachstehenden Beispiele zeigen die verschiedenen 1D-Barcodes mit verschiedenen Optionen. Einige der Optionen sind für verschiedene Codes verfügbar, werden jedoch nur anhand eines Beispiels gezeigt.

Die sich unterhalb des betreffenden Codesymbols befindenden Textfelder erlauben die Eingabe eigener Texte. Dabei sind natürlich die zulässigen Zeichen zu beachten.

Hinweis: *Es kann sein, dass die Barcodes in den Viewer-Ansichten nicht vollständig wiedergegeben werden. Dieser Effekt ist technisch bedingt, und mit stärkerem Zoom-Faktor erscheinen die Balken vollständig und korrekt. Für eine saubere Darstellung der Barcodes empfiehlt es sich zudem, die Dokumente auszudrucken.*

In den Beispielen sind Blinddaten eingegeben. Es kann daher vorkommen, dass die effektiven Werte für den betreffenden Barcode unzulässig sind. Beim Einlesen der Barcodes können Lesefehler auftreten.

Im Dokument code39_family.qdf (siehe Fig. 4-105 (Entwurfsmodus)) bzw. Fig. 4-106 (Ansicht in Snapform Viewer)) sind verschiedene

Beispiele von 1D-Barcodes der Code 39-Gruppe (Code 39, Code 39 Extended, Code 93, Code 93 Extended) gezeigt.

Nach dem Selektieren der Feldelemente im Snapform Designer gibt das Eigenschaften-Fenster Auskunft über dessen jeweiligen Merkmale. Ein Doppelklick auf das Feld öffnet den Formeleditor zur Anzeige der Formel (sofern vorhanden).

Fig. 4-105 *Beispiele von code39_family.qdf, im Entwurfs-Modus 070*

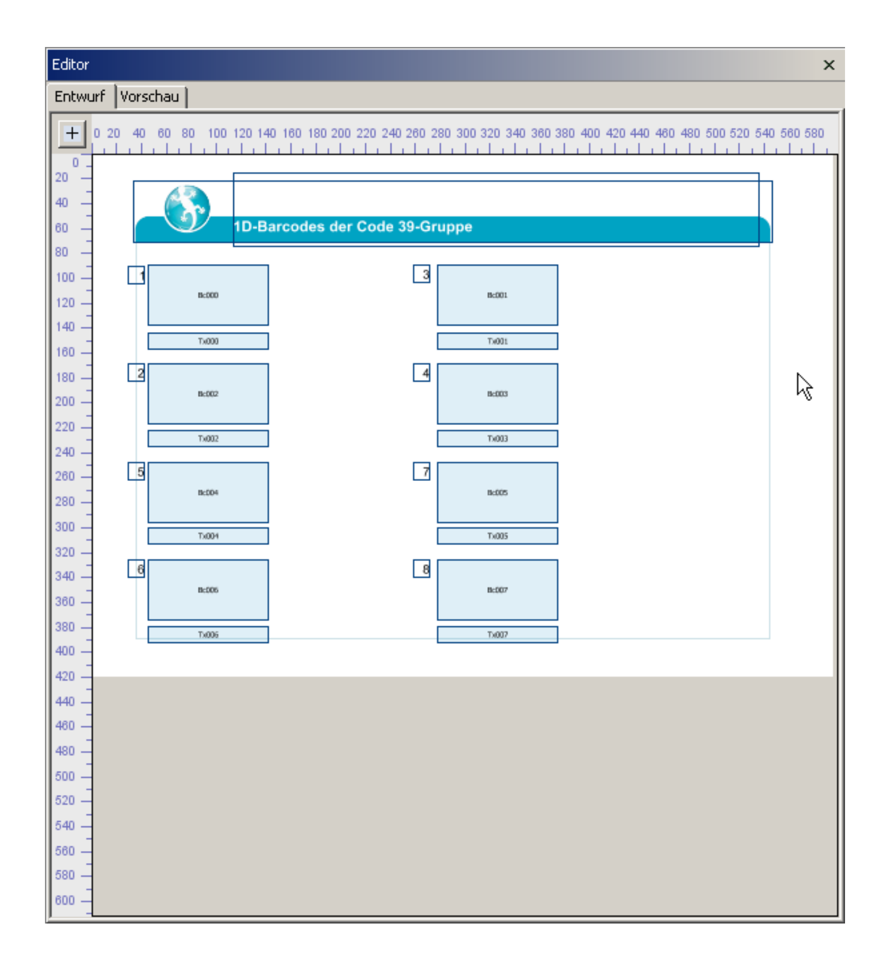
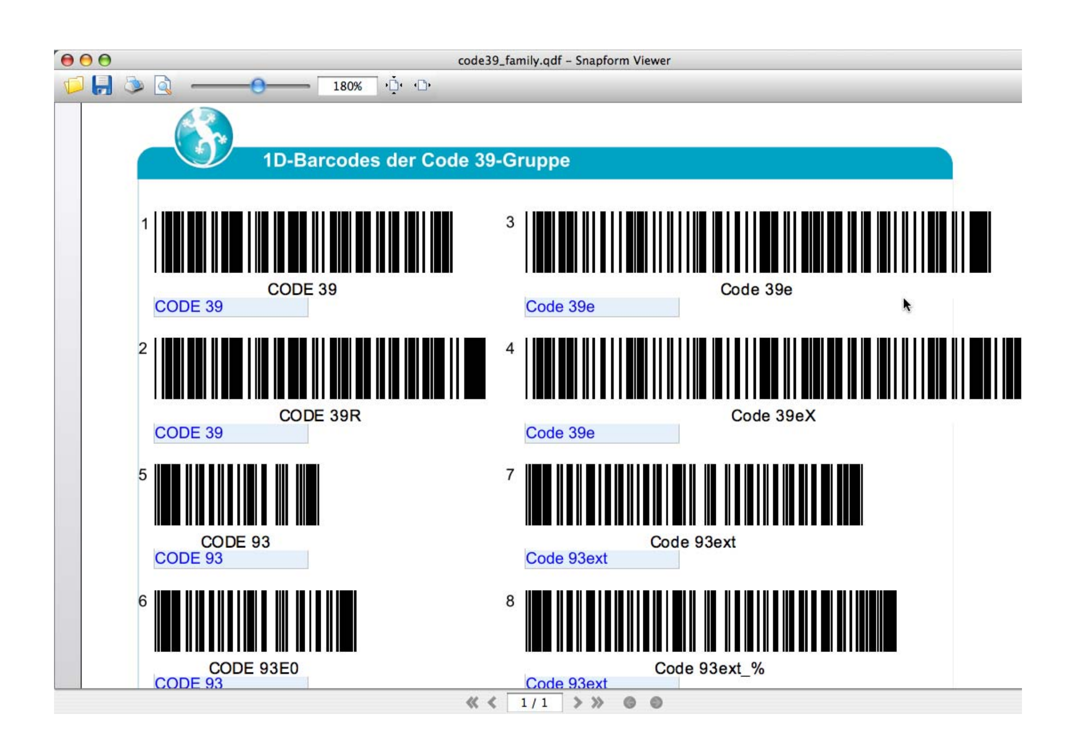


Fig. 4-106 *Beispiele von code39_family.qdf in Snapform Viewer 071*



1 Bc000: Code 39, Text **CODE 39**, ohne Prüfziffer
2 Bc002: Code 39, Text **CODE 39**, mit Prüfziffer
3 Bc001: Code 39 Extended, Text **Code 39e**, ohne Prüfziffer.
4 Bc003: Code 39 Extended, Text **Code 39e**, mit Prüfziffer
5 Bc004: Code 93, Text **CODE 93**, ohne Prüfziffer
6 Bc006: Code 93, Text **CODE 93**, mit Prüfziffer
7 Bc005: Code 93 Extended, Text **Code 93ext**, ohne Prüfziffer
8 Bc007: Code 93 Extended, Text **Code 93ext**, mit Prüfziffer

Im Dokument numeric_family.qdf (siehe Fig. 4-107 (Ansicht in Snapform Viewer) sind verschiedene Beispiele von numerischen 1D-Barcodes (2 of 5, 2 of 5 interleaved, Code 11, Codabar, Code 39 Extended, Code 93, Code 93 Extended) gezeigt.

Hinweis: *Die Ansicht im Entwurfsmodus ist identisch mit Fig. 4-105.*

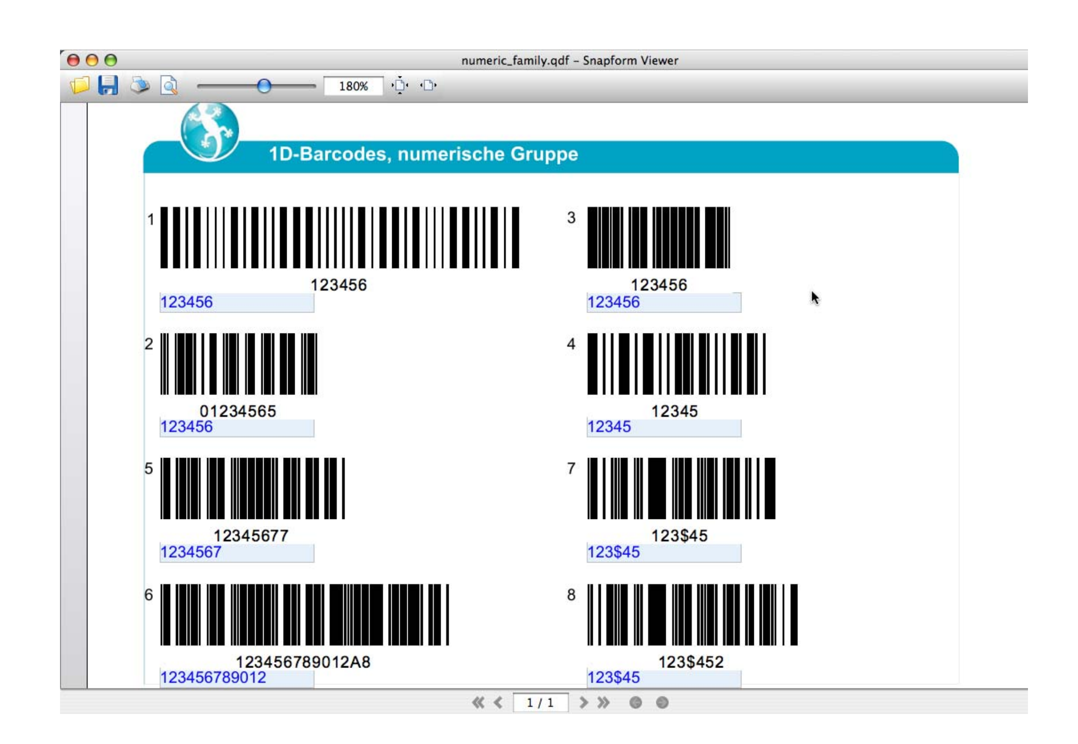


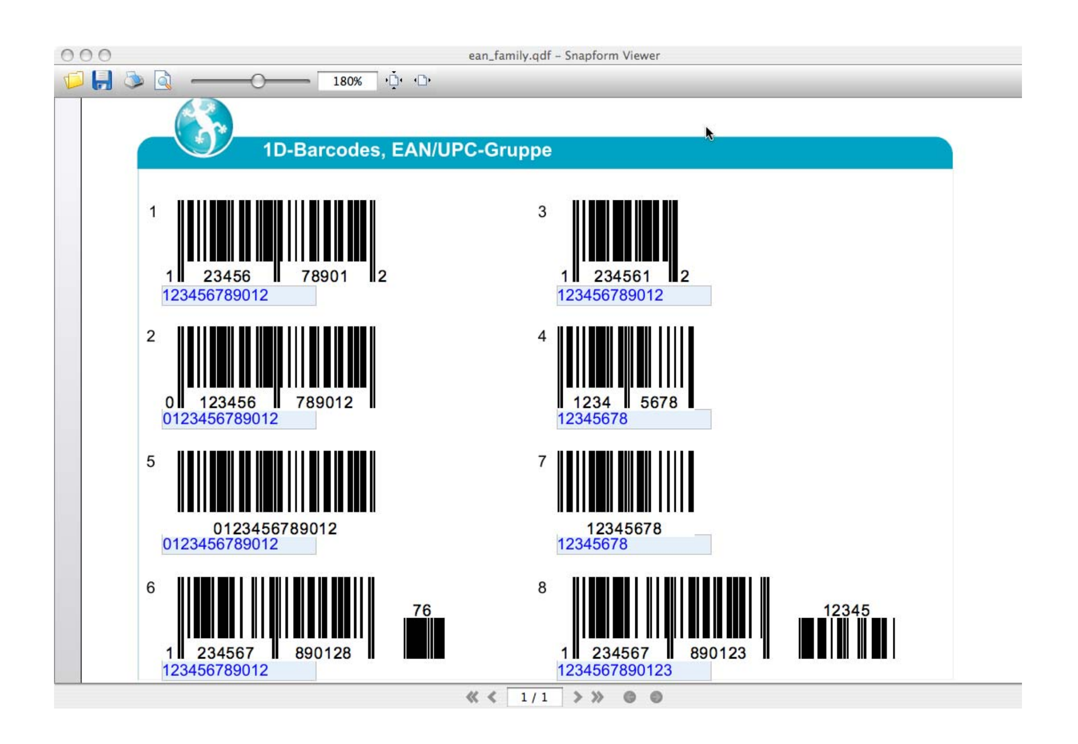
Fig. 4-107 *Beispiele von numeric_family.qdf in Snapform Viewer 072*

1 Bc000: 2 of 5 Code, Text **123456**, ohne Prüfziffer
2 Bc002: interleaved 2 of 5 Code, Text **123456**, mit Prüfziffer
3 Bc001: MAT 25 Code, Text **123456**, ohne Prüfziffer.
4 Bc003: MSI Code, Text **12345**, ohne Prüfziffer
5 Bc004: Code 11, kurzer Text **1234567**, mit Prüfziffer
6 Bc006: Code 11, langer Text **123456789012**, mit Prüfziffer
7 Bc005: Codabar, Text **123$45**, ohne Prüfziffer, Start **A**, Stop **B**
8 Bc007: Codabar, Text **123$45**, mit Prüfziffer, Start **C**, Stop **C**

Im Dokument ean_family.qdf (siehe Fig. 4-108 (Ansicht in Snapform Viewer) sind verschiedene Beispiele von Detailhandels-Barcodes (EAN-8, EAN-13, UPC-A, UPC-E) gezeigt.

Hinweis: *Die Ansicht im Entwurfsmodus ist identisch mit Fig. 4-105.*

Fig. 4-108 *Beispiele von ean_family.qdf in Snapform Viewer 073*

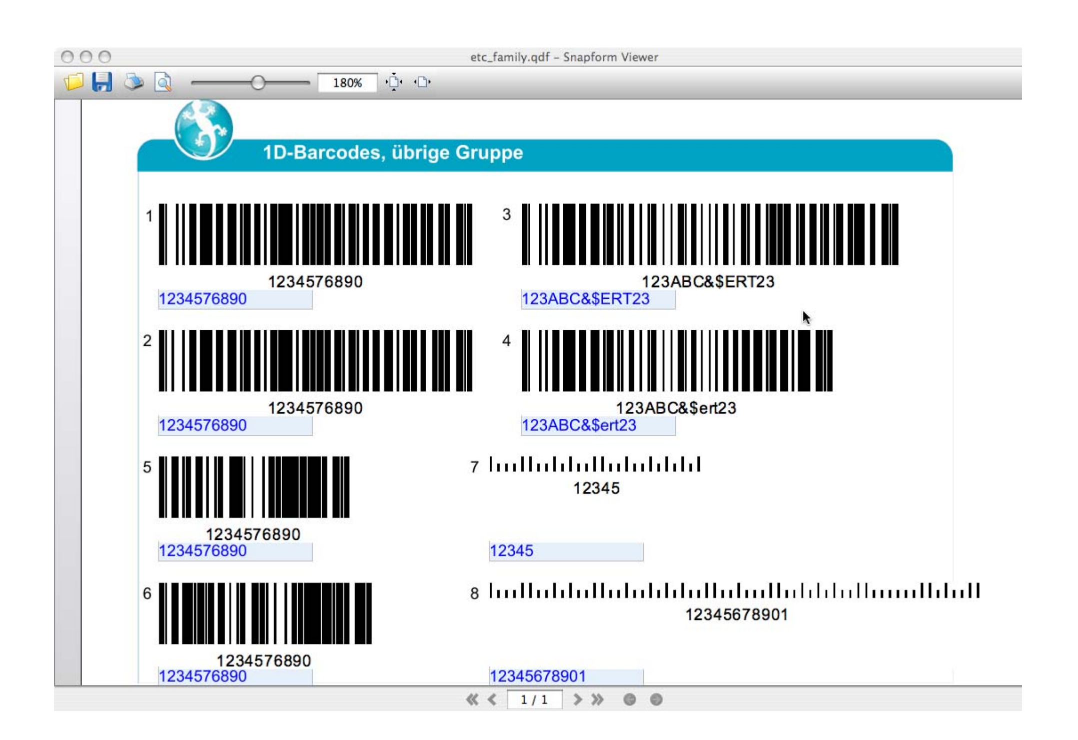


1 Bc000: UPC-A Code, Text **123456789012**, mit Prüfziffer
2 Bc002: EAN-13 Code, Text **0123456789012**, mit Prüfziffer
3 Bc001: UPC-E Code, Text **123456789012**, mit Prüfziffer; die Reduktion der Stellenzahl erfolgt automatisch
4 Bc003: EAN-8 Code, Text **12345678**, mit Prüfziffer
5 Bc004: EAN-13 Code, Text **0123456789012**, mit Prüfziffer, ohne Schutzbalken
6 Bc006: EAN-13 Code, Text **123456789012**, mit Prüfziffer, mit 2-stelligem Zusatzcode **76**
7 Bc005: EAN-8 Code, Text **123$45678**, mit Prüfziffer, ohne Schutzbalken
8 Bc007: EAN-13 Code, Text **123$4567890123**, mit Prüfziffer, mit 5-stelligem Zusatzcode **12345**

Im Dokument etc_family.qdf (siehe Fig. 4-109 (Ansicht in Snapform Viewer) sind verschiedene Beispiele von weiteren Barcodes (Code 128, EAN-128, PostNet) gezeigt.

Hinweis: *Die Ansicht im Entwurfsmodus ist identisch mit Fig. 4-105.*

Fig. 4-109 *Beispiele von etc_family.qdf in Snapform Viewer 074*

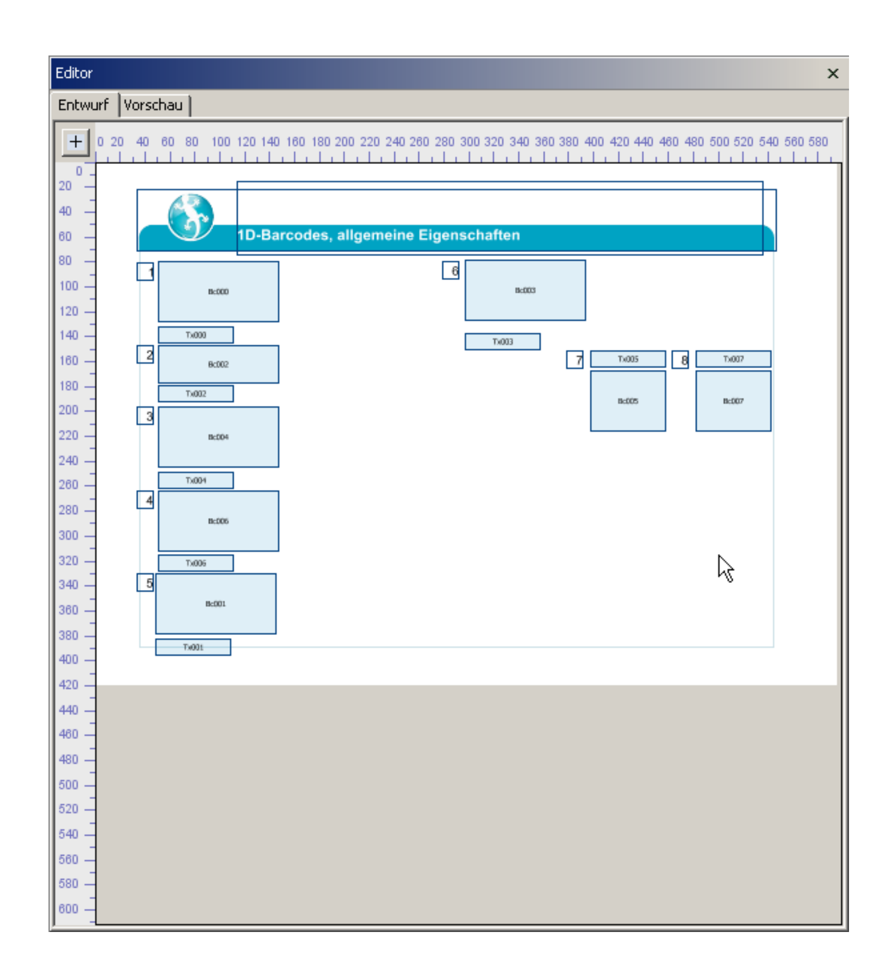


1 Bc000: Code 128, Text **1234567890**, mit Prüfziffer, Zeichensatz A
2 Bc002: Code 128, Text **1234567890**, mit Prüfziffer, Zeichensatz B
3 Bc001: Code 128, Text **123ABC&$ERT23**, mit Prüfziffer; Start-Zeichensatz A
4 Bc003: Code 128, Text **123ABC&$ert23**, mit Prüfziffer, Start-Zeichensatz A
5 Bc004: Code 128, Text **1234567890**, mit Prüfziffer, Zeichensatz C
6 Bc006: EAN-128 Code, Text **1234567890**, mit Prüfziffer, Zeichensatz C
7 Bc005: PostNet Code, Text **12345**, mit Prüfziffer
8 Bc007: PostNet Code, Text **12345678901**, mit Prüfziffer

Im Dokument 1dbarcodes_general.qdf (siehe Fig. 4-110 (Entwurfsmodus)) bzw. Fig. 4-111 (Ansicht in Snapform Viewer)) sind verschiedene Eigenschaften von 1D-Barcodes gezeigt. Als Beispiel wird ein Code 39 benutzt, dessen Wert immer derselbe ist.

Nach dem Selektieren der Feldelemente im Snapform Designer gibt das Eigenschaften-Fenster Auskunft über dessen jeweiligen Merkmale. Ein Doppelklick auf das Feld öffnet den Formeleditor zur Anzeige der Formel (sofern vorhanden).

Fig. 4-110 *Beispiele von 1dbarcodes_general.qdf, im Entwurfs-Modus 075*



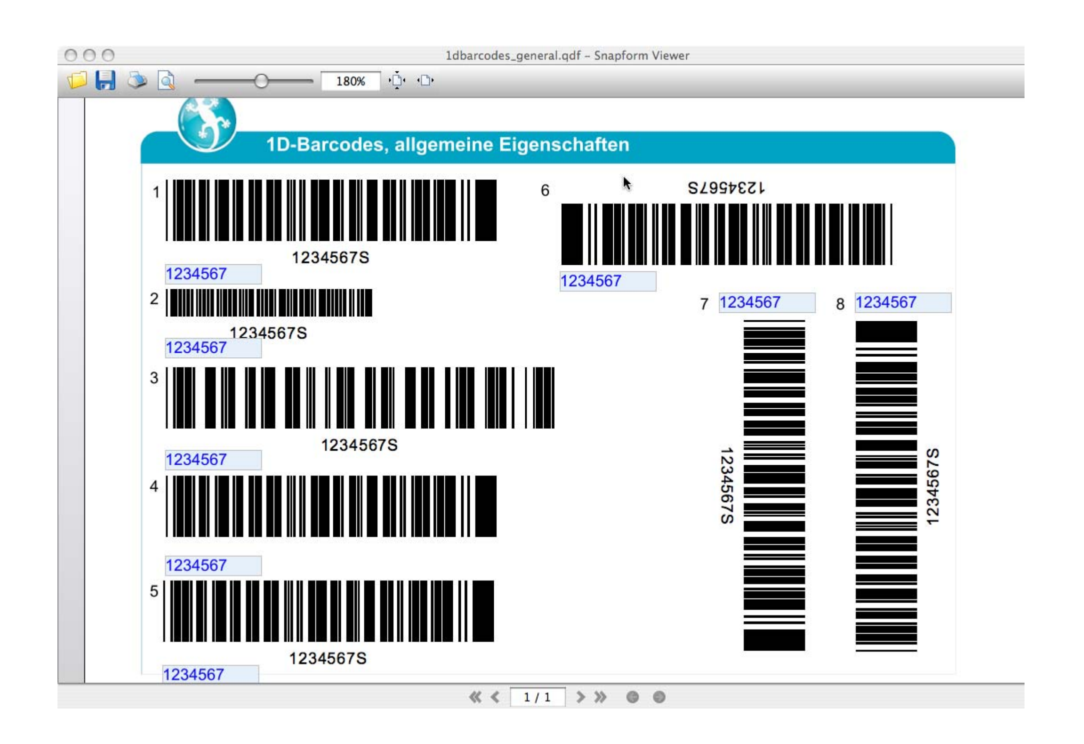


Fig. 4-111 *Beispiele von 1dbarcodes_general.qdf in Snapform Viewer 076*

1 Bc000: Code 39, Text **1234567**, mit Prüfziffer, Einheitenbreite 0,04 cm, Höhe 1 cm, breite Einheit 3 Einheitenbreiten, Trennlücke 2 Einheitenbreiten
2 Bc002: Code 39, Text **1234567**, mit Prüfziffer, Einheitenbreite 0,03 cm, Höhe 0,45 cm, breite Einheit 2 Einheitenbreiten, Trennlücke 1 Einheitenbreiten (diese Parameter sind die Grundeinstellung von Snapform Designer)
3 Bc004: Code 39, Text **1234567**, mit Prüfziffer, Einheitenbreite 0,04 cm, Höhe 1 cm, breite Einheit 3 Einheitenbreiten, Trennlücke 5 Einheitenbreiten
4 Bc006: Code 39, Text **1234567**, mit Prüfziffer, wie Pos 1, jedoch ohne Text
5 Bc001: Code 93, Text **1234567**, mit Prüfziffer, Drehwinkel 0”
6 Bc003: Code 93, Text **1234567**, mit Prüfziffer, Drehwinkel 180”
7 Bc005: Code 93, Text **1234567**, mit Prüfziffer, Drehwinkel 90°
8 Bc007: Code 93, Text **1234567**, mit Prüfziffer, Drehwinkel 270°

4.4.6 2D-Barcodes

2D-Barcodes sind Barcode-Symbologien, welche die Information nicht nur in einer Richtung codieren, sondern in zwei Richtungen. Dadurch

erhöht sich die Kapazität des Symbols um ein vielfaches gegenüber 1D-Barcodes (bis zu 2400 Bytes Binärdaten pro Symbol, was bei entsprechender Kompression bis zu 4000 Zeichen entsprechen kann). Dadurch wird es möglich, den Dateninhalt von Formularen auf kleinem Raum zu codieren, und den sonst bei Papier-basierten Arbeitsabläufen unvermeidlichen Medienbruch zu verhindern.

Zudem gibt es Barcodesysteme, welche eine Verkettung von Symbolen erlauben, und dadurch die verfügbare Datenmenge weiter erhöhen können.

Für die Wahl des Barcodesystems sind die genauen Umstände der Anwendung noch viel stärker in Betracht zu ziehen, als bei 1D-Barcodes. Eine Diskussion der für die Auswahl relevanten Punkten würde den Rahmen dieser Anleitung sprengen. Es wird daher auf die entsprechende Literatur und die Unterstützung durch die Lieferanten von Barcodesystemen verwiesen.

4.4.6.1 Allgemeines zu 2D-Barcodes

Für die Benutzung von 2D-Barcodes in Snapform Designer sind die folgenden Bemerkungen hilfreich.

Bei 2D-Barcodes setzt sich das Symbol aus kleinsten Elementen, genannt Module, zusammen. Module sind meist quadratisch (Seitenverhältnis 1:1). Bei einzelnen Symbologien kann das Seitenverhältnis jedoch kontrolliert werden.

Aufgrund der grösseren Datenmengen und der unterschiedlichen Lesemechanismen reichen Prüfziffern für die Sicherstellung der Integrität der Daten nicht aus. Anstelle dessen werden Fehlerkorrektur-Mechanismen eingebaut, welche auch bei Lesefehlern oder beschädigten Symbolen die Daten noch korrekt auslesen können. Dabei geht eine höhere Sicherheit (höherer Fehlerkorrektur-Grad) zu Lasten der Datenmenge, da die Information zur Fehlerkorrektur in zusätzlichen Daten enthalten ist. Bei entsprechend hohem

Fehlerkorrektur-Grad kann bis zur Hälfte des Symbols beschädigt sein, und die Daten können trotzdem ausgelesen werden.

Die kleinste Informationseinheit ist ein sogenanntes Codewort. Ein Codewort entspricht häufig einem 8-bit Byte. Die Codierung der Daten erfolgt bei Snapform durch die Software und braucht daher für die Formularentwicklung nicht weither in Betracht gezogen werden. Allerdings ist es zur Abschätzung der verfügbaren Datenmenge (und daraus auch der benötigten und verfügbaren Datenmenge für die Fehlerkorrektur) wichtig, die Struktur der Daten zu kennen, um allfälligen Overhead (Steuerzeichen etc.) zu reduzieren.

Snapform unterstützt vier weit verbreitete 2D-Barcode-Systeme: Aztec, PDF-417, Datamatrix und QRCode. Da diese Codes sehr unterschiedlich aufgebaut sind, wurde auf ein gemeinsames Werkzeug verzichtet, und statt dessen für jeden dieser Barcodes ein eigenes Werkzeug implementiert. Diese Werkzeuge und die dazu gehörigen Barcodes werden in den folgenden Abschnitten genauer beschrieben.

4.4.6.2 Aztec Barcode

Der Aztec Barcode ist eine quadratische Matrix-Symbologie hoher Dichte. Die Datenkapazität beträgt 3832 numerische Zeichen, 3067 alphanumerische Zeichen oder 1914 Bytes Binärdaten. Erste Testmuster dieses Codes erinnerten an mittelamerikanische Pyramiden, und dies führte zur Namensgebung.

Die codierten Daten liegen in “Schichten” (quadratischen Ringen) um ein Zentrumssymbol. Diese Anordnung führt zu 32 verschiedenen Symbolgrössen, zwischen 15 x 15 Modulen und 151 x 151 Modulen. Schichten werden immer mit Fehlerkorrekturinformationen aufgefüllt.

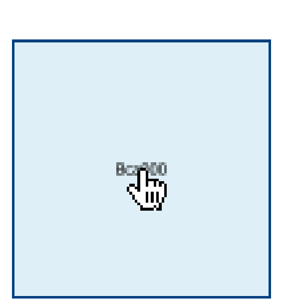
Das Zentrumssymbol ist bei der Bildauswertung ein herausragendes Merkmal, und kann daher leicht erkannt werden. Zusätzliche definierte Module an den Ecken des Zentrumssymbols bestimmen die Orientierung, was es erlaubt, das Symbol aus irgend einer Richtung zu lesen. Da der Code von innen her erzeugt wird, wird keine eigentliche

Ruhezone (weisser Raum um das Symbol) benötigt. Wenn auch die äusserste Schicht nicht zuverlässig gelesen werden könnte, erlaubt die Fehlerkorrektur immer noch eine zuverlässige Decodierung der Daten.

Zeichen werden gemäss ASCII (für Wert 0 bis 127) und gemäss ISO 8859-1, Latin Alphabet 1 (Wert 128 bis 255) codiert. Dies erlaubt die Codierung von Texten in den meisten westlichen Sprachen.

Das Aztec Barcode-Werkzeug erzeugt ein Feld für die Anzeige eines Aztec 2D-Barcodes. Nach Wahl des Werkzeugs erscheint ein Cursor mit einem 2D-Barcodefeld in Default-Grösse (siehe Fig. 4-112).

Fig. 4-112 *Aztec Barcode-Werkzeug-Cursor*



Mit dem Aztec Barcode-Werkzeug verknüpft ist ein 2D-Barcodefeld für Aztec-Code, 96 pt breit und 96 pt hoch. Mit Mausklick wird dieses Feld auf der Arbeitsfläche platziert. Dabei erhält es automatisch einen Namen, bestehend aus den Zeichen Bca und einer dreistelligen fortlaufenden Nummer, beginnend mit 000. Das erste in einem Formular mit dem Aztec Barcode-Werkzeug platzierte Element trägt daher den Namen Bca000. In der Objektstruktur wird es der betreffenden Seite hinzugefügt. In Fig. 4-113 ist das erste Aztec Barcode-Feld in einem neuen Formular platziert und anschliessend mit Anklicken selektiert worden.

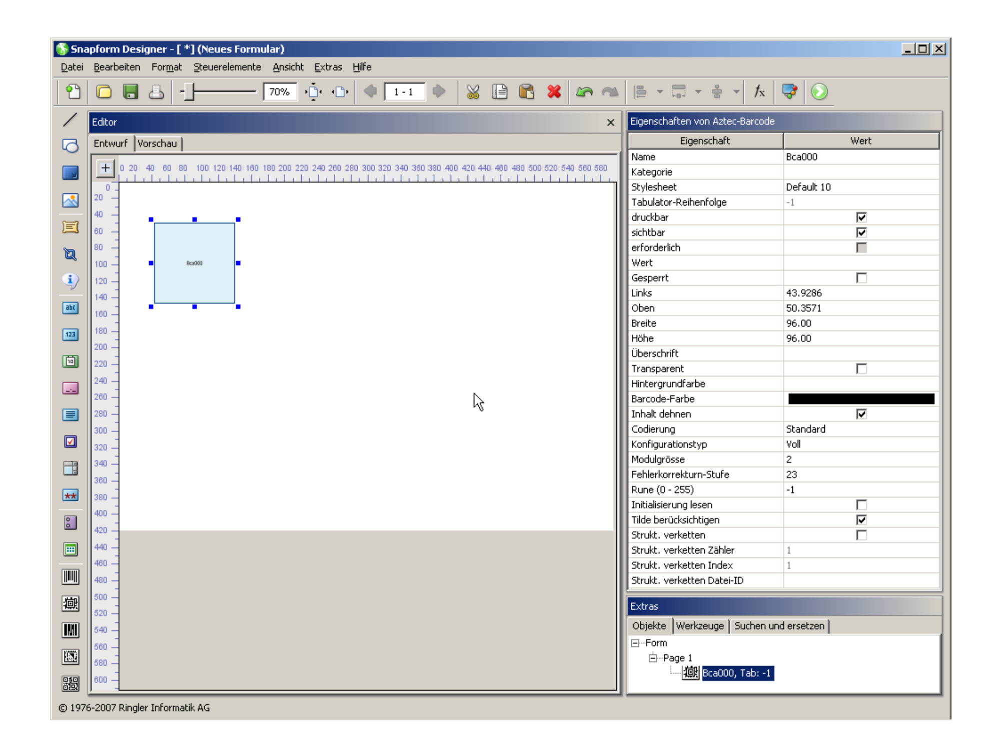


Fig. 4-113 *Neu erstelltes Aztec Barcode-Feld 077*

Die Länge und Breite des Aztec Barcode-Felds kann durch Ziehen der Ankerpunkte des Selektionsrechtecks verändert werden, die Position auch durch Anfassen und Ziehen. Beim Überfahren der Ankerpunkte ändert der Cursor in Doppelpfeile, welche die Bewegungsrichtung des Ankerpunktes angeben. Beim “Anfassen” und Drücken der Maustaste ändert sich der Cursor zu einem Hohlpfeil.

Die übrigen Merkmale eines Aztec Barcode-Feldes werden über das Eigenschaften-Fenster kontrolliert (siehe Fig. 4-114).

Fig. 4-114 *Eigenschaften eines Aztec Barcodefelds 077*

Eigenschaften von Aztec-Barcode

Eigenschaft	Wert
Name	Bca000
Kategorie	
Stylesheet	Default 10
Tabulator-Reihenfolge	-1
druckbar	☑
sichtbar	☑
erforderlich	☐
Wert	
Gesperrt	☐
Links	43.9286
Oben	50.3571
Breite	96.00
Höhe	96.00
Überschrift	
Transparent	☐
Hintergrundfarbe	
Barcode-Farbe	
Inhalt dehnen	☑
Codierung	Standard
Konfigurationstyp	Voll
Modulgrösse	2
Fehlerkorrekturn-Stufe	23
Rune (0 - 255)	-1
Initialisierung lesen	☐
Tilde berücksichtigen	☑
Strukt. verketten	☐
Strukt. verketten Zähler	1
Strukt. verketten Index	1
Strukt. verketten Datei-ID	

Name

Der Name des Aztec Barcodefelds im Dokument. Dieser Name kann geändert werden.

Kategorie

Zugehörigkeit zu einer oder mehreren Kategorien, welche es erlauben, logisch zusammengehörende Gruppen von Elementen zu bilden. Beim Anklicken dieses Felds öffnet sich der Kategorien Dialog (siehe Fig. 4-23).

Die Kategorien werden durch Anklicken des betreffenden Kontrollkästchens ausgewählt. Mit **OK** wird die Auswahl übernommen.

Stylesheet

Das für das Element gültige Stylesheet. Wenn explizite kein Stylesheet ausgewählt worden ist, wird das erste Default-Stylesheet genutzt.

Tabulator-Reihenfolge

Ordnungsnummer des Elements in der Tabulator-Reihenfolge. Dieser Wert ist immer -1 und kann nicht verändert werden. Das bedeutet, dass das Element nicht Bestandteil der Tabulatoren-Reihenfolge ist, was an sich logisch ist, da ein Barcode nicht ein interaktives Element ist.

druckbar

Wenn das Kontrollkästchen gewählt ist, wird das Aztec Barcodefeld ausgedruckt.

sichtbar

Wenn das Kontrollkästchen gewählt ist, wird das Aztec Barcodefeld auf dem Bildschirm angezeigt.

erforderlich

Diese Eigenschaft ist bei Aztec Barcodefeldern immer ausgeschaltet, und kann nicht geändert werden.

Wert

In dieser Eigenschaft des Aztec Barcodefels wird der zu codierende Text eingegeben. Mit Anklicken dieser Eigenschaft öffnet sich der Formeleditor zur Eingabe. In den allermeisten Fällen ist der Wert das Resultat einer Berechnung.

Gesperrt

Wenn das Kontrollkästchen gewählt ist, wird das Aztec Barcodefeld im Snapform Designer blockiert, und kann nicht (versehentlich) verschoben werden. Zudem werden sämtliche übrigen Eigenschaften im Eigenschaften-Dialog deaktiviert. Der Zugriff auf die Eigenschaften ist erst wieder möglich, wenn dieses Kontrollkästchen deaktiviert wird.

Links

X-Koordinate der linken oberen Ecke der Umrandungsbox des Elements in Points. Diese Eigenschaft erlaubt eine genaue numerische Positionierung des Aztec Barcodefelds.

Oben

Y-Koordinate der linken oberen Ecke der Umrandungsbox des Elements in Points. Diese Eigenschaft erlaubt eine genaue numerische Positionierung des Aztec Barcodefelds.

Breite

Breite der Umrandungsbox des Aztec Barcodefelds.

Höhe

Höhe der Umrandungsbox des Aztec Barcodefelds.

Überschrift

Dies ist eine Textzeile, welche oberhalb des Aztec Barcodefelds erscheint. Diese Überschrift kann beispielsweise Informationen für den Sachbearbeiter enthalten, welcher den Code manuell einliest.

Die Überschrift kann entweder als einfacher Text, oder als HTML-Text eingegeben werden. Sie ist von fixer Schriftgrösse (10 pt) und ihre Länge ist auf die Breite des Aztec Barcodefelds beschränkt.

Transparent

Der Hintergrund des Aztec Barcodefelds wird mit diesem Kontrollkästchen auf transparent (keine deckende Hintergrundsfarbe) geschaltet. Diese Eigenschaft ist in der Grundeinstellung ausgeschaltet.

Hintergrundfarbe

Farbe des Barcodefeldes. Mit Anklicken der Eigenschaft öffnet sich das Farbauswahlfenster (siehe Abschnitt 3.2.9.2), welches die Definition der gewünschten Farbe erlaubt.

Die Hintergrundfarbe wird nur dann berücksichtigt, wenn das Kontrollkästchen **Transparent** ausgeschaltet ist.

Barcodefarbe

Farbe des Barcodes. Mit Anklicken der Eigenschaft öffnet sich das Farbauswahlfenster (siehe Abschnitt 3.2.9.2), welches die Definition der gewünschten Farbe erlaubt.

Hinweis: *Die Normen für den Aztec Barcode können Vorschriften bezüglich der zu verwendenden Farben enthalten, und sollten konsultiert werden. Ebenso ist zu beachten, dass einzig die Vollfarben Schwarz, Cyan, Magenta und Yellow vollflächig gedruckt werden (es sei denn, es würden Schmuckfarben eingesetzt), und dass eine gute Druckqualität nur mit diesen Farben bei 100% Sättigung zu nicht aufgerasterten Modulen führt.*

Inhalt dehnen

Dieses Kontrollkästchen bestimmt, ob das Symbol auf die Grösse des Aztec Barcodefelds skaliert wird. Wenn es eingeschaltet ist, wird das Symbol immer auf die Grösse des Felds gebracht. Dabei wird der Wert für die Eigenschaft Modulgrösse automatisch übersteuert. Wenn es

ausgeschaltet ist, wird das Symbol mit der festgelegten Modulgrösse erzeugt und im Zentrum des Felds positioniert.

Vorsicht: *Wenn diese Eigenschaft eingeschaltet ist, kann bei grösseren Datenmengen das Symbol derart verkleinert werden, dass es nicht korrekt ausgedruckt werden kann, und dass die resultierende Modulgrösse zu klein wird.*

Hinweis: *Wenn diese Eigenschaft ausgeschaltet ist, kann das Symbol grösser als das Feld werden, und eventuell andere Elemente des Dokuments überdecken.*

Codierung

Diese Eigenschaft bestimmt, auf welche Art der im Symbol darzustellende Datenstrom codiert wird. Wenn sie gewählt wird, öffnet sich ein Drop Down Menu mit den Optionen **Standard** und **Binär**. **Standard** ist der Grundwert. Mit der Einstellung **Standard** werden Zeichencodes unterhalb 128 (also ASCII) optimal codiert. Andererseits ist die Codierung von Zeichencodes oberhalb 128 nicht sehr effizient. Mit der Einstellung **Binär** werden Zeichencodes nicht optimiert. Diese Einstellung sollte nur verwendet werden, wenn mehrere Zeichen mit Code über 127 zu erwarten sind, da sie die Kapazität des Codes massiv verringert.

Konfigurationstyp

Bei kleinen Datenmengen (bis zu 53 Bytes) kann eine kompakte Form des Aztec Barcodes benutzt werden, welche ein kleineres Zentrumssymbol besitzt und maximal 4 Daten-Schichten aufweisen darf. Ein derartiges Symbol ist kleiner und sollte auch schneller ausgewertet werden können.

Wenn diese Eigenschaft ausgewählt wird, öffnet sich ein Drop Down Menu mit den drei Optionen **Automatisch**, **Voll**, **Kompakt**. **Automatisch** bedeutet, dass das Symbol bei geringen Datenmengen in der kompakten Form dargestellt wird, und dann mit zunehmender Datenmenge automatisch auf die normale Form umschaltet. **Voll** bedeutet, dass das Symbol immer in der normalen grossen Form

dargestellt wird (auch bei geringen Datenmengen. **Kompakt** bedeutet, dass das Symbol immer kompakt dargestellt wird.

Vorsicht: *Wenn bei der gewählten Option* **Kompakt** *die Datenmenge zu gross wird, wird das Symbol als graue Fläche dargestellt.*

Modulgrösse

Dies ist das Mass eines Moduls (der kleinsten Einheit des Aztec Barcodes). Die Masseinheit ist Point, und der Grundwert ist 2. Der Wert sollte nicht kleiner sein, da sonst Probleme beim Druck entstehen könnten.

Hinweis: *Wenn die Eigenschart Dehnen gewählt ist, wird der Wert für die Modulgrösse übersteuert.*

Fehler Korrekturniveau

Dies ist der minimale Anteil an Fehlerkorrekturinformation, welcher im Symbol codiert wird in Prozent der gesamten Informationen. Der Grundwert ist 23%. Der Grundwert reicht für die meisten Applikationen aus, und ermöglicht die korrekte Erfassung, auch wenn bis zu einem Viertel des Symbols falsch gelesen wird. Zulässige Werte liegen zwischen 5% und 95%.

Rune

Es ist möglich, eine Reihe von speziellen Symbolen zu definieren, welchen auf Applikationsebene eine bestimmte Bedeutung zugewiesen werden kann. Ein derartiges Symbol umfasst eine Steuersequenz und eine Ordnungsnummer zwischen 0 und 255. Diese Art Symbol wird Rune genannt.

Der Grundwert für diese Eigenschaft ist **-1**, welcher “ausgeschaltet” entspricht. Zulässige Werte sind ganze Zahlen zwischen **0** und **255**. Wenn ein zulässiger Wert gewählt worden ist, wird dieser codiert, und ein allfälliger Wert der Eigenschaft **Wert** wird ignoriert. Zudem wird automatisch die kompakte Version des Symbols erzeugt (auch wenn bei der Eigenschaft Konfigurationstyp **Voll** gewählt worden ist.

Initialisierung lesen

Wenn dieses Kontrollkästchen gewählt ist, enthält das Aztec Barcodesymbol das sogenannte Initialisierungsflag für die Leseeinheit (den Scanner), welche gewisse Parameter in der Leseeinheit zurücksetzt. Grundstellung dieser Eigenschaft ist ausgeschaltet.

Wiederholungszeiche n ausführen

Zur Anzeige von über die Tastatur nicht darstellbaren Steuerzeichen wird für die Codierung beim Aztec Barcode das Tilde-Zeichen (**~**) benutzt. Diese Eigenschaft bestimmt, ob das Tilde-Zeichen als Steuerzeichen interpretiert wird.

Wenn dieses Kontrollkästchen gewählt wird, wird die Tilde als Steuerzeichen betrachtet. Dies hat die folgenden Konsequenzen, welche bei der Berechnung des zu codierenden Datenstroms zu berücksichtigen sind:

- **~~** wird ersetzt durch das eigentliche Tilde-Zeichen (also Tilde verdoppeln, wenn tatsächlich eine Tilde in den Datenstrom einfliessen muss).
- **~dxxx** wird ersetzt durch das Zeichen, dessen ASCII-Code **xxx** lautet (das heisst, zum Beispiel, dass **~d065** durch **A** ersetzt wird). Diese Darstellung ist wichtig, wenn Steuerzeichen im untersten Bereich codiert werden müssen.
- **~F** wird ersetzt durch das **FNC1** Zeichen. Dies ist ein spezielles Zeichen, welches beim Aztec Barcode-Datenstrom an erster Stelle liegen kann. Diese Zeichenersetzung erfolgt nur am Anfang des Datenstroms.
- **~Exxxxxx** wird ersetzt durch das sogenannte Extended Interpretation Channel flag mit Nummer **xxxxxx**, einem Steuerzeichen, welches auf Applikations-Ebene ausgewertet werden kann.

In der Grundeinstellung ist diese Eigenschaft deaktiviert.

Strukt. verketten

Der Aztec Barcode erlaubt es, bis zu 26 Symbole miteinander zu verketten, und dadurch eine entsprechend höhere Datenkapazität zu erreichen. Beim Erfassen verketteter Symbole speichert die Leseeinheit die Daten und fügt sie zusammen. Erst wenn alle Symbole eingelesen sind, werden die Daten ausgegeben. Gegenüber der nachgeschalteten Applikation erscheint das Ganze, als ob ein riesiges Symbol eingelesen worden wäre.

Wenn dieses Kontrollkästchen gewählt ist, ist das betreffende Symbol Bestandteil einer strukturell verketteten Gruppe. Im anderen Fall ist das Symbol eigenständig.

Wenn diese Eigenschaft gewählt ist, müssen die Eigenschaften **Strukt. verketten Zähler** und **Strukt.verketten Index** einen Wert aufweisen; sie werden im Eigenschaftenfenster auch aktiviert.

Strukt verketten Zähler

Diese Eigenschaft gibt an, wieviele Symbole in der strukturell verketteten Gruppe vorhanden sind, deren Bestandteil das betreffende Symbol ist.

Gültige Werte liegen zwischen **2** und **26**. Der Wert **1** bedeutet, dass die Gruppe aus einen Symbol besteht, und das Symbol daher eigenständig ist.

Diese Eigenschaft ist normalerweise deaktiviert, wird aktiviert, wenn das Kontrollkästchen **Strukt. verketten** eingeschaltet wird.

Strukt verketten Index

Diese Eigenschaft ist die Ordnungsnummer des Symbols in der strukturell verketteten Gruppe.

Gültige Werte liegen zwischen **1** und **26**.

Diese Eigenschaft ist normalerweise deaktiviert. Sie wird aktiviert, wenn das Kontrollkästchen **Strukt. verketten** eingeschaltet wird.

Strukt. verketten Datei-ID

Es ist möglich, den Dateinamen für die ausgelesenen verketteten Daten im voraus zu definieren. Dieser Dateiname wird in dieser Eigenschaft festgelegt.

Diese Eigenschaft ist normalerweise deaktiviert. Sie wird aktiviert wenn das Kontrollkästchen **Strukt. verketten** eingeschaltet wird. Sie ist jedoch optional und wird daher nur applikationsbedingt benötigt.

Im Dokument aztec.qdf (siehe Fig. 4-115 (Entwurfsmodus)) bzw. Fig. 4-116 (Ansicht in Snapform Viewer)) sind verschiedene Beispiele von Aztec Barcodes gezeigt.

Nach dem Selektieren der Barcodefelder im Snapform Designer gibt das Eigenschaften-Fenster Auskunft über deren jeweiligen Merkmale. Ein Doppelklick auf das Feld öffnet den Formeleditor zur Anzeige der hinterlegten Formel.

Fig. 4-115 *Beispiele für Aztec Barcodes von aztec.qdf, im Entwurfs-Modus 078*

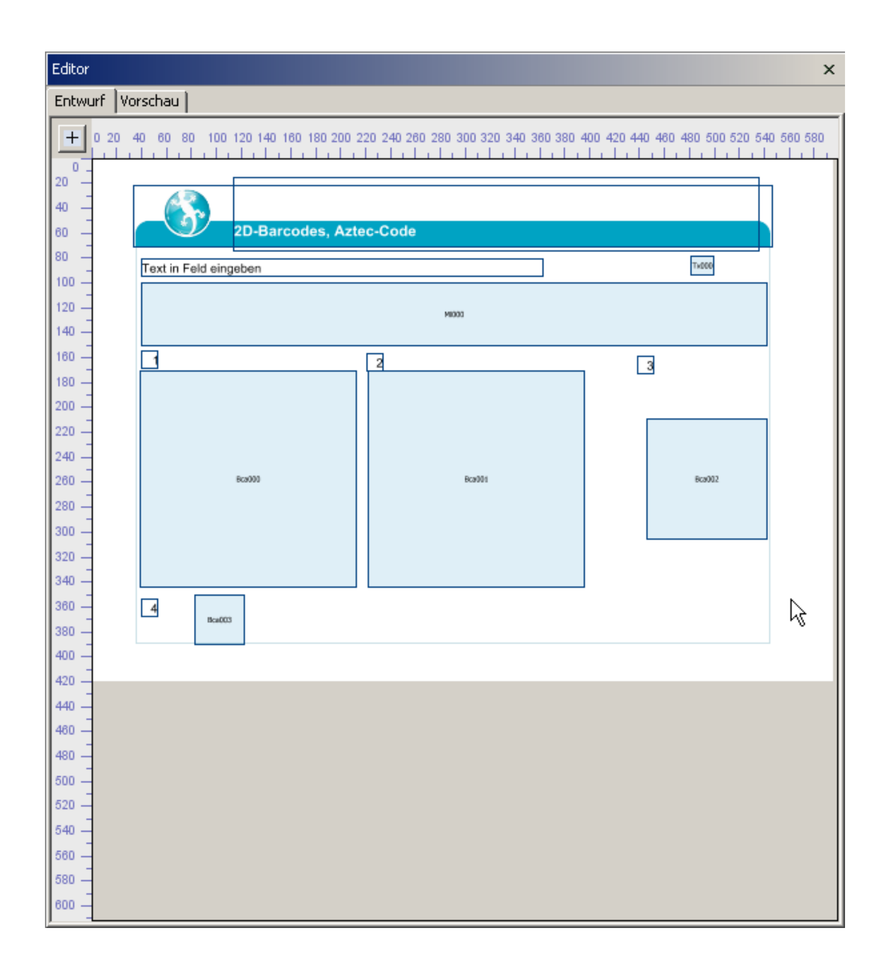
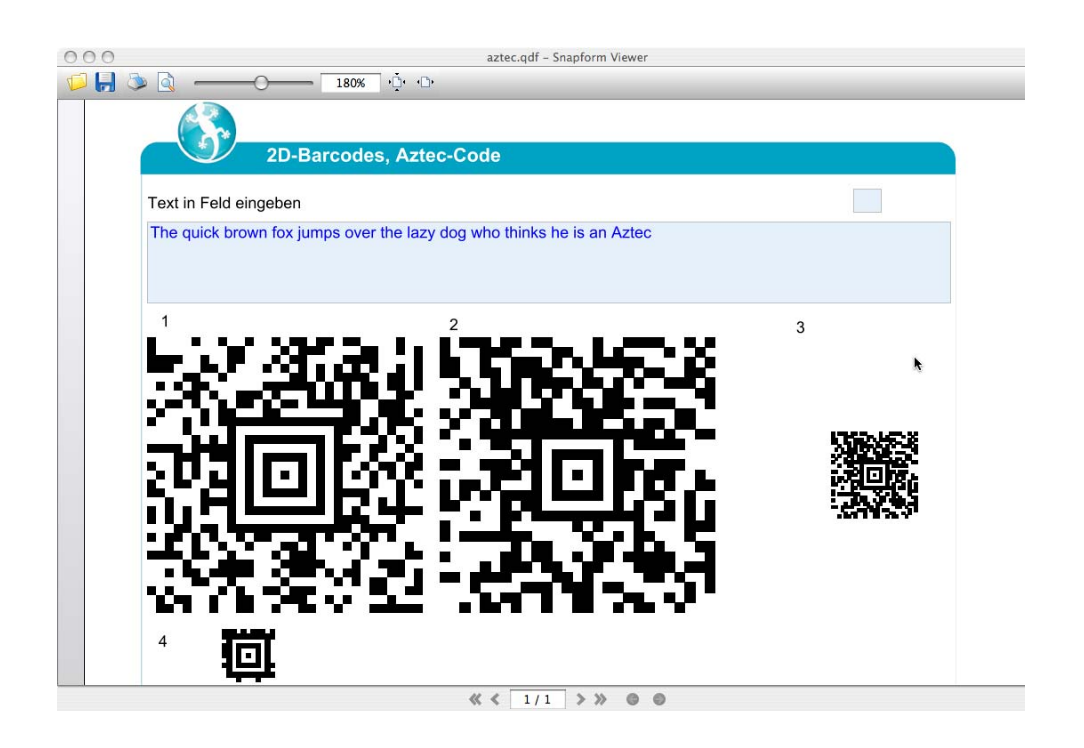


Fig. 4-116 *Beispiele für Aztec Barcodes von aztec.qdf in Snap-form Viewer 079*



1 Bca000: Aztec Barcode, Konfiguration voll, Dehnen ein
2 Bca001: Aztec Barcode, Konfiguration kompakt, Dehnen ein
3 Bca002: Aztec Barcode, Konfiguration Automatik, Dehnen aus, kleineres Feld
4 Bca003: Aztec Barcode, Konfiguration Automatik, Dehnen ein, Rune 4

Alle vier Barcodes stellen den im Eingabefeld definierten Text entsprechend den gewählten Eigenschaften dar. In der Beispieldatei kann der Text verändert werden, und die Symbole verändern sich entsprechend.

4.4.6.3 PDF417 Barcode

Der PDF-417 Barcode ist eine 2-dimensionale “geschichtete” Symbologie hoher Dichte. Die Datenkapazität beträgt rund 1800 Zeichen bzw. 1100 binäre Zeichen pro Symbol.

Die codierten Daten liegen in Form von Zeilen vor, welche mit einem Startzeichen beginnen, eine gewisse Anzahl Datenzeichen enthalten und mit einem Stopzeichen enden. Daraus ergibt sich die typische

Spaltenstruktur. Die Anzahl Datenzeichen kann variieren, wie auch die Anzahl Zeilen (mindestens 3, maximal 90).

Die Daten werden in sogenannte Codewörter wiedergegeben. Ein Codewort besteht aus vier Balken und vier Lücken, welche auf einen Raum von 17 Modulen aufgeteilt sind.

Mit zunehmender Datenmenge wächst das Symbol in Spaltenrichtung, indem weitere Zeilen hinzugefügt werden. Bei der Planung eines Formulars muss daher entsprechend Freiraum vorgesehen werden. Ebenso muss um das Symbol eine Ruhezone vorhanden sein, damit es einwandfrei gelesen werden kann.

PDF-417 hat verschiedene Stufen von Fehlerkorrektur. Mit der höchsten Einstellung (maximale Anzahl Fehlerkorrektur-Codewörter) kann zwar nicht mehr die maximale Datenkapazität erreicht werden, aber ein zur Hälfte abgedecktes oder sonstwie verändertes Symbol kann noch gelesen werden.

Ein weiteres Merkmal des PDF-417 Barcodes ist die Möglichkeit der Verkettung von einer grossen Anzahl Symbolen zur Darstellung von wirklich grossen Datenmengen. Diese Eigenschaft wird Makro-PDF-417 genannt.

Das PDF-417 Barcode-Werkzeug erzeugt ein Feld für die Anzeige eines PDF-417 2D-Barcodes. Nach Wahl des Werkzeugs erscheint ein Cursor mit einem 2D-Barcodefeld in Default-Grösse (siehe Fig. 4-117).

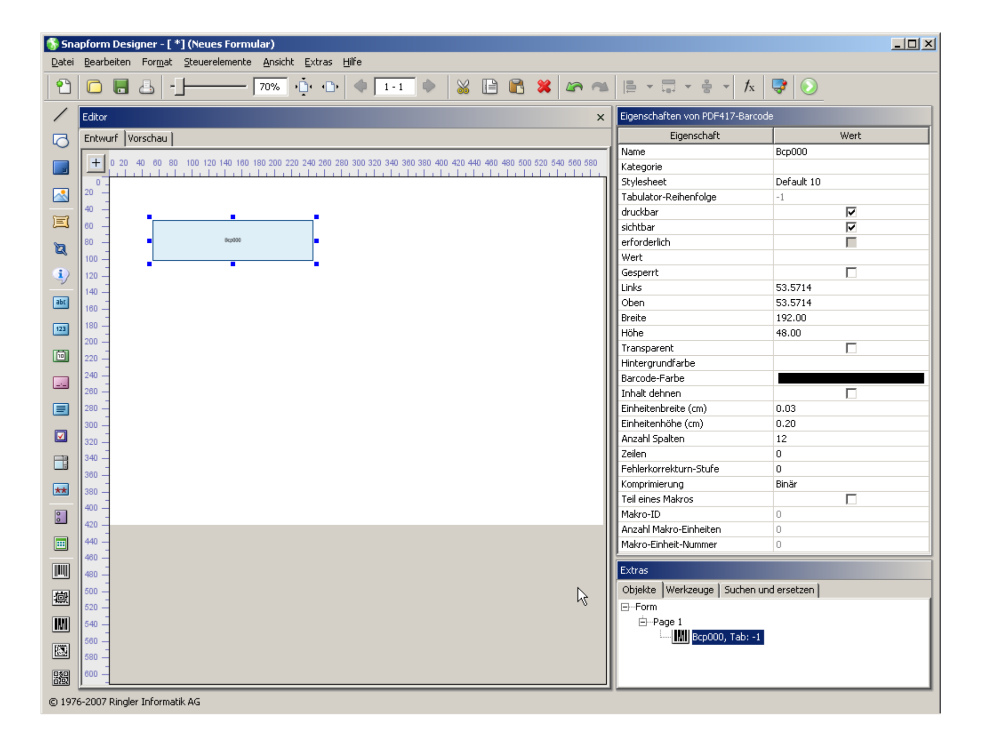
Fig. 4-117 *PDF-417 Barcode-Werkzeug-Cursor*



Mit dem PDF-417 Barcode-Werkzeug verknüpft ist ein 2D-Barcodefeld für PDF-417-Code, 192 pt breit und 48 pt hoch. Mit Mausklick wird dieses Feld auf der Arbeitsfläche platziert. Dabei erhält es automatisch einen Namen, bestehend aus den Zeichen Bcp und einer dreistelligen

fortlaufenden Nummer, beginnend mit 000. Das erste in einem Formular mit dem PDF-417 Barcode-Werkzeug platzierte Element trägt daher den Namen Bcp000. In der Objektstruktur wird es der betreffenden Seite hinzugefügt. In Fig. 4-118 ist das erste PDF-417 Barcode-Feld in einem neuen Formular platziert und anschliessend mit Anklicken selektiert worden.

Fig. 4-118 *Neu erstelltes PDF-417 Barcode-Feld 080*



Die Länge und Breite des PDF-417 Barcode-Felds kann durch Ziehen der Ankerpunkte des Selektionsrechtecks verändert werden, die Position auch durch Anfassen und Ziehen. Beim Überfahren der Ankerpunkte ändert der Cursor in Doppelpfeile, welche die Bewegungsrichtung des Ankerpunktes angeben. Beim "Anfassen" und Drücken der Maustaste ändert sich der Cursor zu einem Hohlpfeil.

Die übrigen Merkmale eines PDF-417 Barcode-Feldes werden über das Eigenschaften-Fenster kontrolliert (siehe Fig. 4-119).

Fig. 4-119 *Eigenschaften eines PDF-417 Barcode-felds 080*

Eigenschaften von PDF417-Barcode

Eigenschaft	Wert
Name	Bcp000
Kategorie	
Stylesheet	Default 10
Tabulator-Reihenfolge	-1
druckbar	☑
sichtbar	☑
erforderlich	☐
Wert	
Gesperrt	☐
Links	53.5714
Oben	53.5714
Breite	192.00
Höhe	48.00
Transparent	☐
Hintergrundfarbe	
Barcode-Farbe	
Inhalt dehnen	☐
Einheitenbreite (cm)	0.03
Einheitenhöhe (cm)	0.20
Anzahl Spalten	12
Zeilen	0
Fehlerkorrekturn-Stufe	0
Komprimierung	Binär
Teil eines Makros	☐
Makro-ID	0
Anzahl Makro-Einheiten	0
Makro-Einheit-Nummer	0

Name

Der Name des PDF-417 Barcodefelds im Dokument. Dieser Name kann geändert werden.

Kategorie

Zugehörigkeit zu einer oder mehreren Kategorien, welche es erlauben, logisch zusammengehörende Gruppen von Elementen zu bilden. Beim Anklicken dieses Felds öffnet sich der Kategorien Dialog (siehe Fig. 4-23).

Die Kategorien werden durch Anklicken des betreffenden Kontrollkästchens ausgewählt. Mit **OK** wird die Auswahl übernommen.

Stylesheet

Das für das Element gültige Stylesheet. Wenn explizite kein Stylesheet ausgewählt worden ist, wird das erste Default-Stylesheet genutzt.

Tabulator-Reihenfolge

Ordnungsnummer des Elements in der Tabulator-Reihenfolge. Dieser Wert ist immer -1 und kann nicht verändert werden. Das bedeutet, dass das Element nicht Bestandteil der Tabulatoren-Reihenfolge ist, was an sich logisch ist, da ein Barcode nicht ein interaktives Element ist.

druckbar

Wenn das Kontrollkästchen gewählt ist, wird das PDF-417 Barcodefeld ausgedruckt.

sichtbar

Wenn das Kontrollkästchen gewählt ist, wird das PDF-417 Barcodefeld auf dem Bildschirm angezeigt.

erforderlich

Diese Eigenschaft ist bei PDF-417 Barcodefeldern immer ausgeschaltet, und kann nicht geändert werden.

Wert

In dieser Eigenschaft des PDF-417 Barcodefels wird der zu codierende Text eingegeben. Mit Anklicken dieser Eigenschaft öffnet sich der Formeleditor zur Eingabe. In den allermeisten Fällen ist der Wert das Resultat einer Berechnung.

Gesperrt

Wenn das Kontrollkästchen gewählt ist, wird das PDF-417 Barcodefeld im Snapform Designer blockiert, und kann nicht (versehentlich) verschoben werden. Zudem werden sämtliche übrigen Eigenschaften im Eigenschaften-Dialog deaktiviert. Der Zugriff auf die Eigenschaften ist erst wieder möglich, wenn dieses Kontrollkästchen deaktiviert wird.

Links

X-Koordinate der linken oberen Ecke der Umrandungsbox des Elements in Points. Diese Eigenschaft erlaubt eine genaue numerische Positionierung des PDF-417 Barcodefelds.

Oben

Y-Koordinate der linken oberen Ecke der Umrandungsbox des Elements in Points. Diese Eigenschaft erlaubt eine genaue numerische Positionierung des PDF-417 Barcodefelds.

Breite

Breite der Umrandungsbox des PDF-417 Barcodefelds.

Höhe

Höhe der Umrandungsbox des PDF-417 Barcodefelds.

Transparent

Der Hintergrund des PDF-417 Barcodefelds wird mit diesem Kontrollkästchen auf transparent (keine deckende Hintergrundsfarbe) geschaltet. Diese Eigenschaft ist in der Grundeinstellung ausgeschaltet.

Hintergrundfarbe

Farbe des Barcodefeldes. Mit Anklicken der Eigenschaft öffnet sich das Farbauswahlfenster (siehe Abschnitt 3.2.9.2), welches die Definition der gewünschten Farbe erlaubt.

Die Hintergrundfarbe wird nur dann berücksichtigt, wenn das Kontrollkästchen **Transparent** ausgeschaltet ist.

Barcodefarbe

Farbe des Barcodes. Mit Anklicken der Eigenschaft öffnet sich das Farbauswahlfenster (siehe Abschnitt 3.2.9.2), welches die Definition der gewünschten Farbe erlaubt.

Hinweis: *Die Normen für den PDF-417 Barcode können Vorschriften bezüglich der zu verwendenden Farben enthalten, und sollten konsultiert werden. Ebenso ist zu beachten, dass einzig die Vollfarben Schwarz, Cyan, Magenta und Yellow vollflächig gedruckt werden (es sei denn, es würden Schmuckfarben eingesetzt), und dass eine gute Druckqualität nur mit diesen Farben bei 100% Sättigung zu nicht aufgerasterten Modulen führt.*

Inhalt dehnen

Dieses Kontrollkästchen bestimmt, ob das Symbol auf die Grösse des PDF-417 Barcodefelds skaliert wird. Wenn es eingeschaltet ist, wird das Symbol immer auf die Grösse des Felds gebracht. Dabei wird die Modulgrösse automatisch übersteuert. Wenn es ausgeschaltet ist, wird das Symbol mit der festgelegten Modulgrösse erzeugt und im Zentrum des Felds positioniert.

Vorsicht: *Wenn diese Eigenschaft eingeschaltet ist, kann bei grösseren Datenmengen das Symbol derart verkleinert werden, dass es nicht korrekt ausgedruckt werden kann, und dass die resultierende Modulgrösse zu klein wird.*

Hinweis: *Wenn diese Eigenschaft ausgeschaltet ist, kann das Symbol grösser als das Feld werden, und eventuell andere Elemente des Dokuments überdecken.*

Einheitenbreite (cm)

Diese Eigenschaft bestimmt die Modulbreite des PDF-417 Barcodes. Sie wird in cm eingegeben. Da der PDF-417 Barcode rechteckige Module zulässt (im Gegensatz zu anderen Systemen, welche quadratische Module voraussetzen), kann dieser Wert von der Einheitenhöhe abweichen. Empfohlen werden jedoch ganz- oder halbzahlige Vielfache.

Der Grundwert beträgt **0.015 cm**. Es wird jedoch empfohlen, nicht unter **0.075 cm** zu gehen.

Hinweis: *Wenn die Eigenschart* **Dehnen** *gewählt ist, wird der Wert für die Einheitenbreite übersteuert.*

Einheitenbreite (cm)

Diese Eigenschaft bestimmt die Modulhöhe des PDF-417 Barcodes. Sie wird in cm eingegeben. Da der PDF-417 Barcode rechteckige Module zulässt (im Gegensatz zu anderen Systemen, welche quadratische Module voraussetzen), kann dieser Wert von der Einheitenbreite abweichen. Empfohlen werden jedoch ganz- oder halbzahlige Vielfache.

Der Grundwert beträgt **0.015 cm**. Es wird jedoch empfohlen, nicht unter **0.075 cm** zu gehen.

Hinweis: *Wenn die Eigenschart* **Dehnen** *gewählt ist, wird der Wert für die Einheitenhöhe übersteuert.*

Anzahl Spalten

Diese Eigenschaft bestimmt, wie breit der PDF-417 Barcode wird. Dieser Wert bestimmt die Anzahl Spalten für Nutzdaten. Schliesslich werden im endgültigen Symbol vier weitere Spalten hinzugefügt,

Der kleinste Wert für diese Eigenschaft beträgt **3**, und der Wert sollte **18** nicht übersteigen, damit das Symbol zuverlässig gelesen werden kann.

Zeilen

Diese Eigenschaft bestimmt, wieviele Zeilen der Code mindestens aufweisen muss. Wenn die Datenmenge geringer ist als notwendig, werden solange Füllzeichen hinzugefügt, bis die Zeilenzahl erreicht wird.

Der kleinste Wert für diese Eigenschaft beträgt **3**, und der Wert sollte **90** nicht überschreiten, damit das Symbol sicher gelesen werden kann.

Wenn der Wert **0** eingegeben wird, werden soviele Zeilen erzeugt, wie notwendig sind.

Drehwinkel

Der Barcode kann um Vielfache von 90° gedreht werden. Beim Anklicken dieser Eigenschaft öffnet sich ein Drop Down Menu mit der Auswahl der Drehwinkel **0°**, **90°**, **180°**, **270°**. Grundeinstellung ist 0°.

Fehler Korrekturniveau

Dies ist der Grad der Fehlerkorrektur. Der eingegebene Wert kann eine ganze Zahl zwischen 0 (keine Fehlerkorrektur) und 8 (maximale Fehlerkorrektur) sein. Dieser Wert bestimmt auch die maximale Menge an Nutzdaten, welche im Symbol codiert werden kann.

Komprimierung

Um ein Maximum an Daten codieren zu können, werden sie komprimiert. Der PDF-417 Barcode kennt drei Kompressionen, welche auf bestimmte Art Daten optimiert sind: Zahlen, Text und Binärdaten. Die grösste Kompression kann erreicht werden, wenn nur Zahlen codiert werden; die geringste für Binärdaten.

Wenn diese Eigenschaft gewählt wird, öffnet sich ein Drop Down Menu mit den Optionen **Binär**, **Text** und **Numerisch**. Grundwert ist **Text**. Dies entspricht den oben genannten Kompressionsmethoden. Im Datenstrom können “fremde” Zeichen enthalten sein. Für detailliertere Angaben zu den Kompressionsmethoden im PDF-417 Barcode wird auf die Literatur verwiesen.

Wenn die Option Binär gewählt ist, muss tatsächlich ein binärer Datenstrom für die Codierung bereit gestellt werden. Das bedeutet, dass “normale” Texte vorkomprimiert werden müssen. Hierzu können

entweder die Funktionen Compress(x) oder getBytes(string) benutzt werden. Im Beispiel wird für Position 4 die folgende Formel benutzt:

```
= Compress(M1000)
```

Hinweis: *Wenn die Daten nicht mit der gewählten Kompressionsart übereinstimmen, wird das Symbol nicht angezeigt. Bei der Fehlersuche sollte daher als erstes diese Eigenschaft kontrolliert werden.*

Teil eines Makros

Der PDF-417 Barcode erlaubt es, bis zu 127 Symbole miteinander zu verketten, und dadurch eine entsprechend höhere Datenkapazität zu erreichen. Diese Sonderform des Barcodes wird Makro-PDF-417 genannt. Beim Erfassen verketteter Symbole speichert die Leseeinheit die Daten und fügt sie zusammen. Erst wenn alle Symbole eingelesen sind, werden die Daten ausgegeben. Gegenüber der nachgeschalteten Applikation erscheint das Ganze, als ob ein riesiges Symbol eingelesen worden wäre.

Wenn dieses Kontrollkästchen gewählt ist, ist das betreffende Symbol Bestandteil einer verketteten Gruppe. Im anderen Fall ist das Symbol eigenständig.

Wenn diese Eigenschaft gewählt ist, müssen die Eigenschaften **Anzahl Makro-Einheiten** und **Makro-Einheit-Nummer** einen Wert aufweisen; sie werden im Eigenschaftenfenster auch aktiviert. Das Feld für die Eigenschaft **Makro-ID** wird ebenfalls aktiviert, ist jedoch nicht zwingend auszufüllen.

Makro-ID

Es ist möglich, den Dateinamen für die ausgelesenen verketteten Daten im voraus zu definieren. Dieser Dateiname wird in dieser Eigenschaft festgelegt.

Diese Eigenschaft ist normalerweise deaktiviert. Sie wird aktiviert wenn das Kontrollkästchen **Teil eines Makros** eingeschaltet wird. Sie ist jedoch optional und wird daher nur applikationsbedingt benötigt.

Anzahl Makro-Einheiten

Diese Eigenschaft gibt an, wieviele Symbole in der verketteten Gruppe vorhanden sind, deren Bestandteil das betreffende Symbol ist.

Gültige Werte liegen zwischen **2** und **127**. Der Wert **1** bzw. **0** bedeutet, dass die Gruppe aus einen Symbol besteht, und das Symbol daher eigenständig ist.

Diese Eigenschaft ist normalerweise deaktiviert, wird aktiviert, wenn das Kontrollkästchen **Teil eines Makros** eingeschaltet wird.

Makro-Einheit-Nummer

Diese Eigenschaft ist die Ordnungsnummer des Symbols in der verketteten Gruppe.

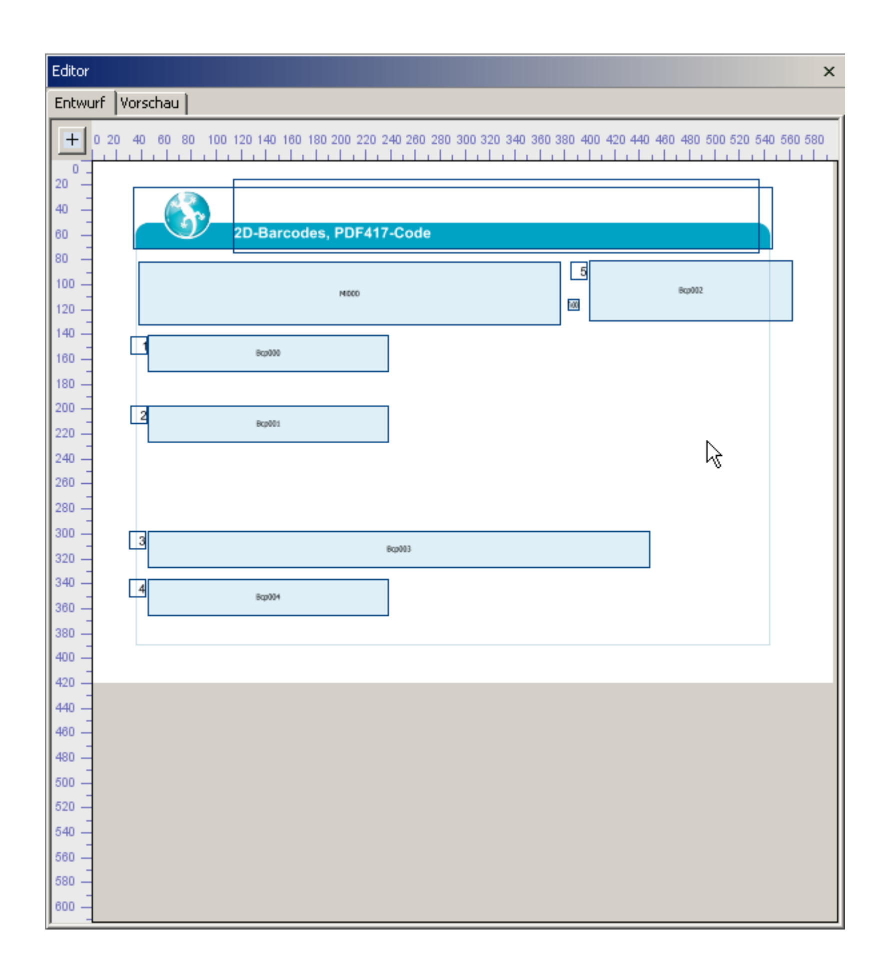
Gültige Werte liegen zwischen **1** und **127**. Der Wert **0** bedeutet, dass das Symbol nicht bestandteil einer verketteten Gruppe ist. Der Wert dieser Eigenschaft muss zudem kleiner sein als der Wert von **Anzahl Makro-Einheiten**.

Diese Eigenschaft ist normalerweise deaktiviert. Sie wird aktiviert, wenn das Kontrollkästchen **Teil eines Makros** eingeschaltet wird.

Im Dokument pdf417.qdf (siehe Fig. 4-120 (Entwurfsmodus)) bzw. Fig. 4-121 (Ansicht in Snapform Viewer)) sind verschiedene Beispiele von PDF-417 Barcodes gezeigt.

Nach dem Selektieren der Barcodefelder im Snapform Designer gibt das Eigenschaften-Fenster Auskunft über deren jeweiligen Merkmale. Ein Doppelklick auf das Feld öffnet den Formeleditor zur Anzeige der hinterlegten Formel.

Fig. 4-120 *Beispiele für PDF-417 Barcodes von pdf417.qdf im Entwurfs-Modus 081*



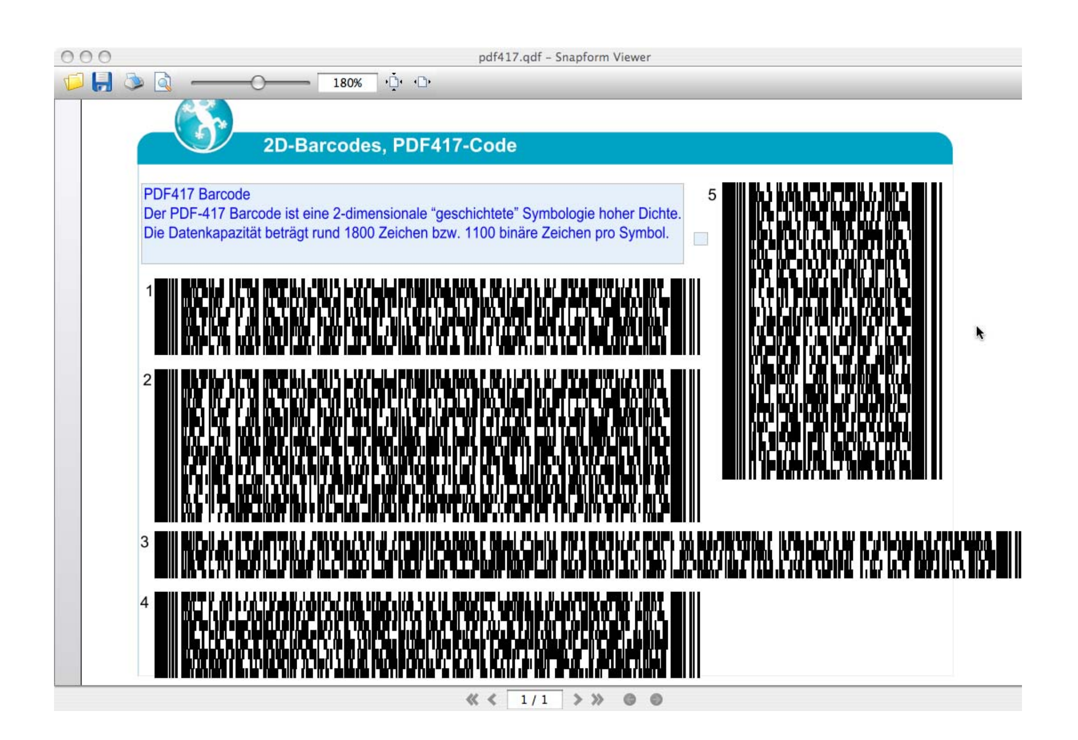


Fig. 4-121 *Beispiele für PDF-417 Barcodes von pdf417.qdf in Snapform Viewer 082*

1 Bcp000: PDF-417 Barcode, 16 Spalten, Fehlerkorrekturgrad 2
2 Bca001: PDF-417 Barcode, 16 Spalten, Fehlerkorrekturgrad 6
3 Bca002: PDF-417 Barcode, 28 Spalten, Fehlerkorrekturgrad 2
4 Bca004: PDF-417 Barcode, 16 Spalten, Fehlerkorrekturgrad 2, Daten vorkomprimiert mit `compress(x)`, Symbol Binär komprimiert
5 Bca003: PDF-417 Barcode, 4 Spalten, Fehlerkorrekturgrad 2

Alle fünf Barcodes stellen den im Eingabefeld definierten Text entsprechend den gewählten Eigenschaften. In der Beispieldatei kann der Text verändert werden, und die Symbole verändern sich entsprechend.

4.4.6.4 Datamatrix

Der Datamatrix Barcode ist eine zweidimensionale Matrix-Symbologie hoher Dichte mit quadratischen Modulen. Die maximale Datenkapazität beträgt 3116 numerische Zeichen, 2335 alphanumerische Zeichen oder 1556 Bytes Binärdaten. Die codierten Daten setzen sich aus quadratischen Bereichen zusammen, und das Symbol selbst ist normalerweise auch quadratisch. Gewisse rechteckige Formate sind jedoch möglich.

Die quadratischen Bereiche werden von einem sogenannten Findermuster umrahmt, welches auch zur Orientierung des Symbols genutzt wird. Das Symbol selbst benötigt eine Ruhezone (leere Umrandung um das Symbol), damit es einwandfrei gelesen werden kann.

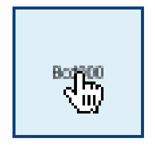
Die Grösse und Ausgestaltung (einschliesslich Fehlerkorrektur-Information) der quadratischen Bereiche sowie der Symbole ist in den Normen festgelegt. Datamatrix-Symbole können nicht dynamisch wachsen; wenn die Datenmenge den für eine bestimmte Symbolgrösse massgebenden Wert überschreitet, muss das nächstgrössere Symbol gewählt werden. In Snapform kann die Wahl der Symbolgrösse automatisch erfolgen.

Der Datamatrix Barcode kann 256 Zeichen des "normalen" ASCII-Zeichensatzes codieren. Mit Hilfe von sogenannten Extended Channels können andere Zeichensätze codiert werden. Ein Wechsel zwischen Zeichensätzen ist mittels entsprechender Steuerzeichen auch innerhalb eines Symbols möglich. Ein derartiger Wechsel muss allerdings applikatorisch unterstützt werden.

Gemäss der Norm für den Datamatrix Barcode können bis zu 16 Symbols strukturell verknüpft werden. Diese Möglichkeit ist jedoch in Snapform nicht implementiert.

Das Datamatrix Barcode-Werkzeug erzeugt ein Feld für die Anzeige eines Datamatrix 2D-Barcodes. Nach Wahl des Werkzeugs erscheint ein Cursor mit einem 2D-Barcodefeld in Default-Grösse (siehe Fig. 4-122).

Fig. 4-122 *Datamatrix Barcode-Werkzeug-Cursor*



Mit dem Datamatrix Barcode-Werkzeug verknüpft ist ein 2D-Barcodefeld für Datamatrix-Code, 48 pt breit und 48 pt hoch. Mit Mausklick wird dieses Feld auf der Arbeitsfläche platziert. Dabei erhält es automatisch einen Namen, bestehend aus den Zeichen Bcd und einer dreistelligen fortlaufenden Nummer, beginnend mit 000. Das erste in einem Formular mit dem Datamatrix Barcode-Werkzeug platzierte Element trägt daher den Namen Bcd000. In der Objektstruktur wird es der betreffenden Seite hinzugefügt. In Fig. 4-123 ist das erste Datamatrix Barcode-Feld in einem neuen Formular platziert und anschliessend mit Anklicken selektiert worden.

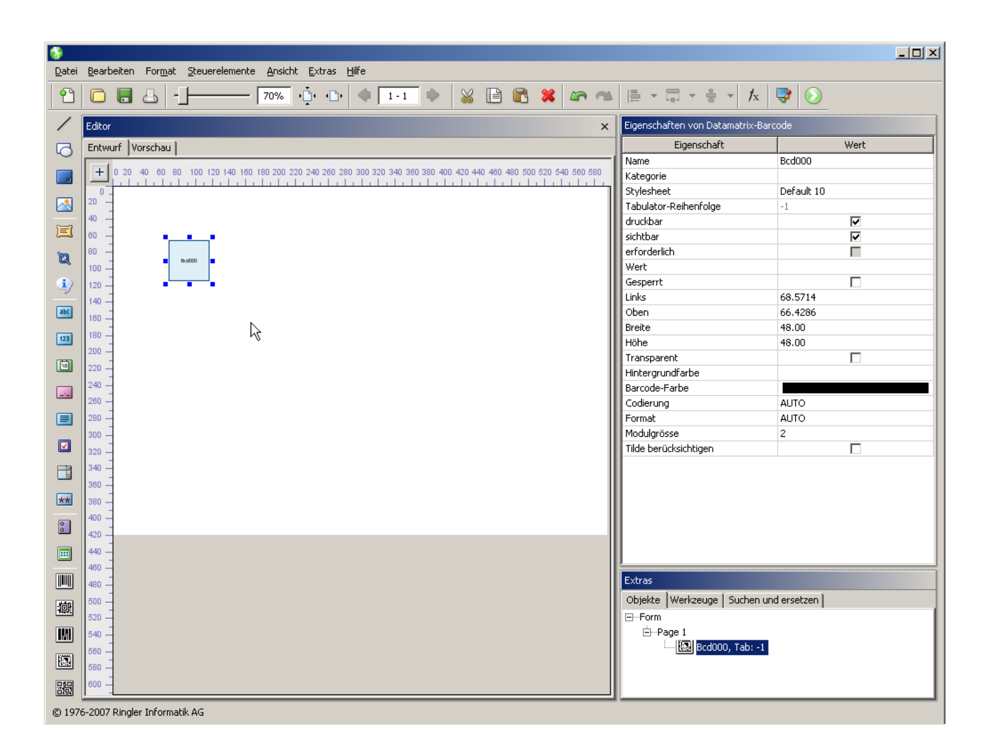


Fig. 4-123 *Neu erstelltes Datamatrix Barcode-Feld 083*

Die Länge und Breite des Datamatrix Barcode-Felds kann durch Ziehen der Ankerpunkte des Selektionsrechtecks verändert werden, die Position auch durch Anfassen und Ziehen. Beim Überfahren der Ankerpunkte ändert der Cursor in Doppelpfeile, welche die Bewegungsrichtung des Ankerpunktes angeben. Beim "Anfassen" und Drücken der Maustaste ändert sich der Cursor zu einem Hohlpfeil.

Hinweis: *Das Feld wird automatisch zu einem Quadrat skaliert, wenn ein Ankerpunkt verschoben wird. Dabei ist die längere Seite massgebend.*

Die übrigen Merkmale eines Datamatrix Barcode-Feldes werden über das Eigenschaften-Fenster kontrolliert (siehe Fig. 4-124).

Fig. 4-124 *Eigenschaften eines Datamatrix Barcodefelds 083*

Eigenschaften von Datamatrix-Barcode

Eigenschaft	Wert
Name	Bcd000
Kategorie	
Stylesheet	Default 10
Tabulator-Reihenfolge	-1
druckbar	☑
sichtbar	☑
erforderlich	☐
Wert	
Gesperrt	☐
Links	68.5714
Oben	66.4286
Breite	48.00
Höhe	48.00
Transparent	☐
Hintergrundfarbe	
Barcode-Farbe	
Codierung	AUTO
Format	AUTO
Modulgrösse	2
Tilde berücksichtigen	☐

Name

Der Name des Datamatrix Barcodefelds im Dokument. Dieser Name kann geändert werden.

Kategorie

Zugehörigkeit zu einer oder mehreren Kategorien, welche es erlauben, logisch zusammengehörende Gruppen von Elementen zu bilden. Beim Anklicken dieses Felds öffnet sich der Kategorien Dialog (siehe Fig. 4-23).

Die Kategorien werden durch Anklicken des betreffenden Kontrollkästchens ausgewählt. Mit **OK** wird die Auswahl übernommen.

Stylesheet

Das für das Element gültige Stylesheet. Wenn explizite kein Stylesheet ausgewählt worden ist, wird das erste Default-Stylesheet genutzt.

Tabulator-Reihenfolge

Ordnungsnummer des Elements in der Tabulator-Reihenfolge. Dieser Wert ist immer -1 und kann nicht verändert werden. Das bedeutet, dass das Element nicht Bestandteil der Tabulatoren-Reihenfolge ist, was an sich logisch ist, da ein Barcode nicht ein interaktives Element ist.

druckbar

Wenn das Kontrollkästchen gewählt ist, wird das Datamatrix Barcodefeld ausgedruckt.

sichtbar

Wenn das Kontrollkästchen gewählt ist, wird das Datamatrix Barcodefeld auf dem Bildschirm angezeigt.

erforderlich

Diese Eigenschaft ist bei Datamatrix Barcodefeldern immer ausgeschaltet, und kann nicht geändert werden.

Wert

In dieser Eigenschaft des Datamatrix Barcodefels wird der zu codierende Text eingegeben. Mit Anklicken dieser Eigenschaft öffnet sich der Formeleditor zur Eingabe. In den allermeisten Fällen ist der Wert das Resultat einer Berechnung.

Gesperrt

Wenn das Kontrollkästchen gewählt ist, wird das Datamatrix Barcodefeld im Snapform Designer blockiert, und kann nicht (versehentlich) verschoben werden. Zudem werden sämtliche übrigen Eigenschaften im Eigenschaften-Dialog deaktiviert. Der Zugriff auf die Eigenschaften ist erst wieder möglich, wenn dieses Kontrollkästchen deaktiviert wird.

Links

X-Koordinate der linken oberen Ecke der Umrandungsbox des Elements in Points. Diese Eigenschaft erlaubt eine genaue numerische Positionierung des Datamatrix Barcodefelds.

Oben

Y-Koordinate der linken oberen Ecke der Umrandungsbox des Elements in Points. Diese Eigenschaft erlaubt eine genaue numerische Positionierung des Datamatrix Barcodefelds.

Breite

Breite der Umrandungsbox des Datamatrix Barcodefelds.

Sofern ihr Wert geringer ist, wird die Höhe automatisch der Breite angepasst, damit das Feld wieder quadratisch ist. Anderenfalls wird der Wert der Breite nicht verändert..

Höhe

Höhe der Umrandungsbox des Datamatrix Barcodefelds.

Sofern ihr Wert geringer ist, wird die Breite automatisch der Höhe angepasst, damit das Feld wieder quadratisch ist. Anderenfalls wird der Wert der Höhe nicht verändert..

Transparent

Der Hintergrund des Datamatrix Barcodefelds wird mit diesem Kontrollkästchen auf transparent (keine deckende Hintergrundsfarbe) geschaltet. Diese Eigenschaft ist in der Grundeinstellung ausgeschaltet.

Hintergrundfarbe

Farbe des Barcodefeldes. Mit Anklicken der Eigenschaft öffnet sich das Farbauswahlfenster (siehe Abschnitt 3.2.9.2), welches die Definition der gewünschten Farbe erlaubt.

Die Hintergrundfarbe wird nur dann berücksichtigt, wenn das Kontrollkästchen **Transparent** ausgeschaltet ist.

Barcodefarbe

Farbe des Barcodes. Mit Anklicken der Eigenschaft öffnet sich das Farbauswahlfenster (siehe Abschnitt 3.2.9.2), welches die Definition der gewünschten Farbe erlaubt.

Hinweis: *Die Normen für den Datamatrix Barcode können Vorschriften bezüglich der zu verwendenden Farben enthalten, und sollten konsultiert werden. Ebenso ist zu beachten, dass einzig die Vollfarben Schwarz, Cyan, Magenta und Yellow vollflächig gedruckt werden (es sei denn, es würden Schmuckfarben eingesetzt), und dass eine gute Druckqualität nur mit diesen Farben bei 100% Sättigung zu nicht aufgerasterten Modulen führt.*

Drehwinkel

Der Barcode kann um Vielfache von 90° gedreht werden. Beim Anklicken dieser Eigenschaft öffnet sich ein Drop Down Menu mit der Auswahl der Drehwinkel **0°**, **90°**, **180°**, **270°**. Grundeinstellung ist 0°.

Codierung

Diese Eigenschaft bestimmt, auf welche Art der im Symbol darzustellende Datenstrom codiert wird. Wenn sie gewählt wird, öffnet sich ein Drop Down Menu mit den Optionen **AUTO**, **ASCII**, **C40**, **TEXT**

und **BASE256**. **AUTO** ist der Grundwert. Die Einstellungen haben folgende Bedeutung:

AUTO
Die bestmögliche Datencodierung wird durch die Software automatisch gewählt.

ASCII
Diese Codierung wird benutzt, wenn der Datenstrom weitgehend aus Zeichen mit ASCII-Werten zwischen 0 und 127 besteht. Sie codiert ein alphanumerisches oder zwei numerische Zeichen in ein Byte.

C40
Diese Codierung wird benutzt, wenn der Datenstrom weitgehend aus Zahlen und Grossbuchstaben besteht. Sie codiert drei alphanumerische Zeichen in zwei Byte.

TEXT
Diese Codierung wird benutzt, wenn der Datenstrom weitgehend aus Zahlen und Kleinbuchstaben besteht. Sie codiert drei alphanumerische Zeichen in zwei Byte.

BASE256
Diese Codierung wird benutzt, wenn der Datenstrom aus 8-bit Binärdaten besteht.

Im Normalfall erzielt die Stellung AUTO ausreichend gut codierte Symbole.

Format
Der Datamatrix Barcode hat eine definierte Reihe von Abmessungen, gemessen in Anzahl Modulen pro Kante. Diese Abmessungen sind bedingt durch die interne Struktur des Symbols. Die nachstehenden Abmessungen sind definiert und zulässig:

1. quadratisch

   10x10, 12x12, 14x14, 16x16, 18x18, 20x20, 22x22, 24x24, 26x26, 32x32, 36x36, 40x40, 44x44, 48x48, 52x52, 64x64, 72x72, 80x80, 88x88, 96x96, 104x104, 120x120, 132x132, 144x144

2. rechteckig

   8x18, 8x32, 12x26, 12x36, 16x36, 16x48

Wenn diese Eigenschaft gewählt wird, öffnet sich ein Drop Down Menu mit den Optionen **AUTO** gefolgt von der Liste der oben genannten Abmessungen. **AUTO** ist der Grundwert.

Die Option **AUTO** bedeutet, dass aufgrund der zu codierenden Daten die optimale Symbolgrösse gewählt wird. Das Symbol ist an der linken oberen Ecke des Barcodefelds verankert, und daher verschiebt sich beim Wachsen dessen linke untere Ecke.

Hinweis: *Auch wenn eine fixe Abmessung gewählt worden ist, wird beim Überschreiten der maximalen Datenmenge das nächstgrössere Format verwendet. Es ist daher wichtig, die zu codierende Datenmenge sorgfältig abzuschätzen und ausreichend Freiraum um das Symbol frei zu halten.*

Hinweis: *Die rechteckigen Symbole haben (wie ersichtlich) eine geringe Datenkapazität. Das grösste rechteckige Symbol (16x48) kann bestenfalls 72 alphanumerische Zeichen bzw. 47 Byte an binären Daten codieren. Wird diese Datenmenge überschritten, wird ein geeignetes quadratisches Format verwendet.*

Modulgrösse

Dies ist das Mass eines Moduls (der kleinsten Einheit des Datamatrix Barcodes). Die Masseinheit ist Point, und der Grundwert ist 2. Der Wert sollte nicht kleiner sein, da sonst Probleme beim Druck entstehen könnten.

Wiederholungszeichen ausführen

Zur Anzeige von über die Tastatur nicht darstellbaren Steuerzeichen wird für die Codierung beim Datamatrix Barcode das Tilde-Zeichen (**~**) benutzt. Diese Eigenschaft bestimmt, ob das Tilde-Zeichen als Steuerzeichen interpretiert wird.

Wenn dieses Kontrollkästchen gewählt wird, wird die Tilde als Steuerzeichen betrachtet. Dies hat die folgenden Konsequenzen,

welche bei der Berechnung des zu codierenden Datenstroms zu berücksichtigen sind:

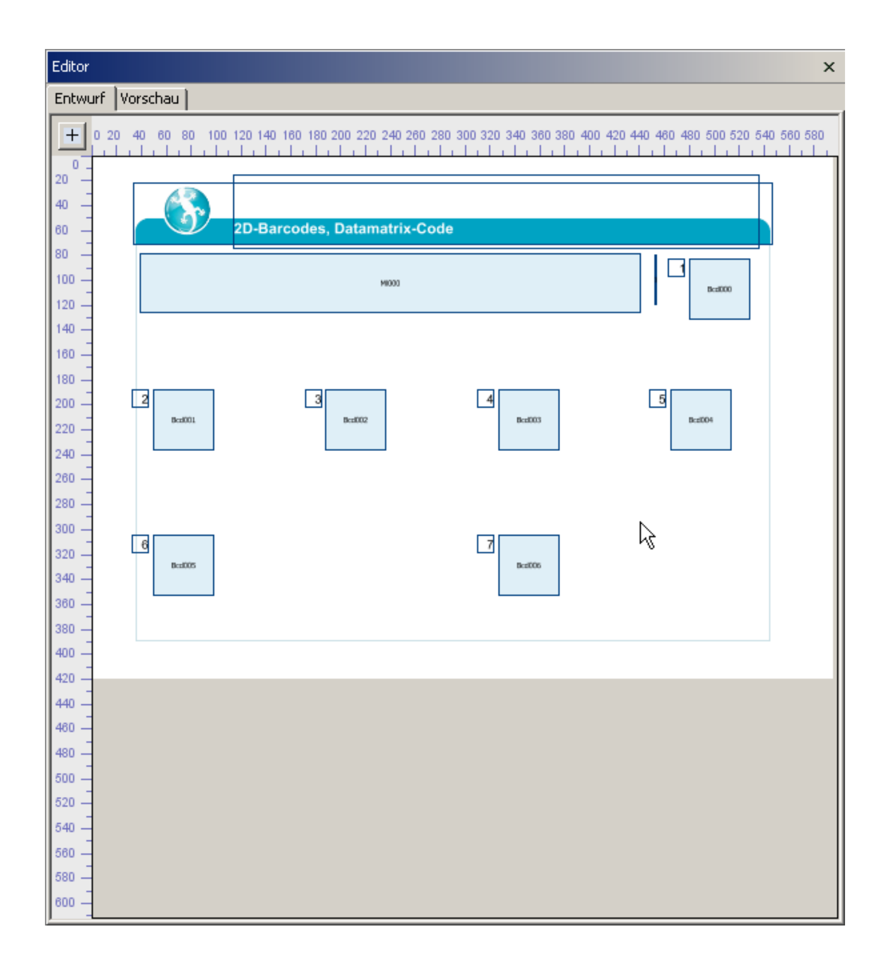
- **~~** wird ersetzt durch das eigentliche Tilde-Zeichen (also Tilde verdoppeln, wenn tatsächlich eine Tilde in den Datenstrom einfliessen muss).
- **~dxxx** wird ersetzt durch das Zeichen, dessen ASCII-Code **xxx** lautet (das heisst, zum Beispiel, dass **~d065** durch **A** ersetzt wird). Diese Darstellung ist wichtig, wenn Steuerzeichen im untersten Bereich codiert werden müssen.
- **~1** wird ersetzt durch das **FNC1** Zeichen. Dies ist ein spezielles Zeichen, welches beim Datamatrix Barcode-Datenstrom an erster Stelle liegen kann und besagt, dass die Daten dem UCC/EAN Standard Identifier Format entsprechen. Dieses Zeichenersetzung erfolgt nur am Anfang des Datenstroms.
- **~3** ist ein spezielles Zeichen, welches beim Datamatrix Barcode-Datenstrom an erster Stelle liegen kann und besagt, dass die Daten Steuerfunktionen für die Leseeinheit enthalten.
- **~Exxxxxx** wird ersetzt durch das sogenannte Extended Interpretation Channel flag mit Nummer **xxxxxx**, einem Steuerzeichen, welches auf Applikations-Ebene ausgewertet werden kann, und häufig zur Anzeige von Zeichensatzwechseln benutzt wird..

In der Grundeinstellung ist diese Eigenschaft deaktiviert.

Im Dokument datamatrix.qdf (siehe Fig. 4-125 (Entwurfsmodus)) bzw. Fig. 4-126 (Ansicht in Snapform Viewer)) sind verschiedene Beispiele von Datamatrix Barcodes gezeigt.

Nach dem Selektieren der Barcodefelder im Snapform Designer gibt das Eigenschaften-Fenster Auskunft über deren jeweiligen Merkmale. Ein Doppelklick auf das Feld öffnet den Formeleditor zur Anzeige der hinterlegten Formel.

Fig. 4-125 *Beispiele für Datamatrix Barcodes von datamatrix.qdf, im Entwurfs-Modus 084*



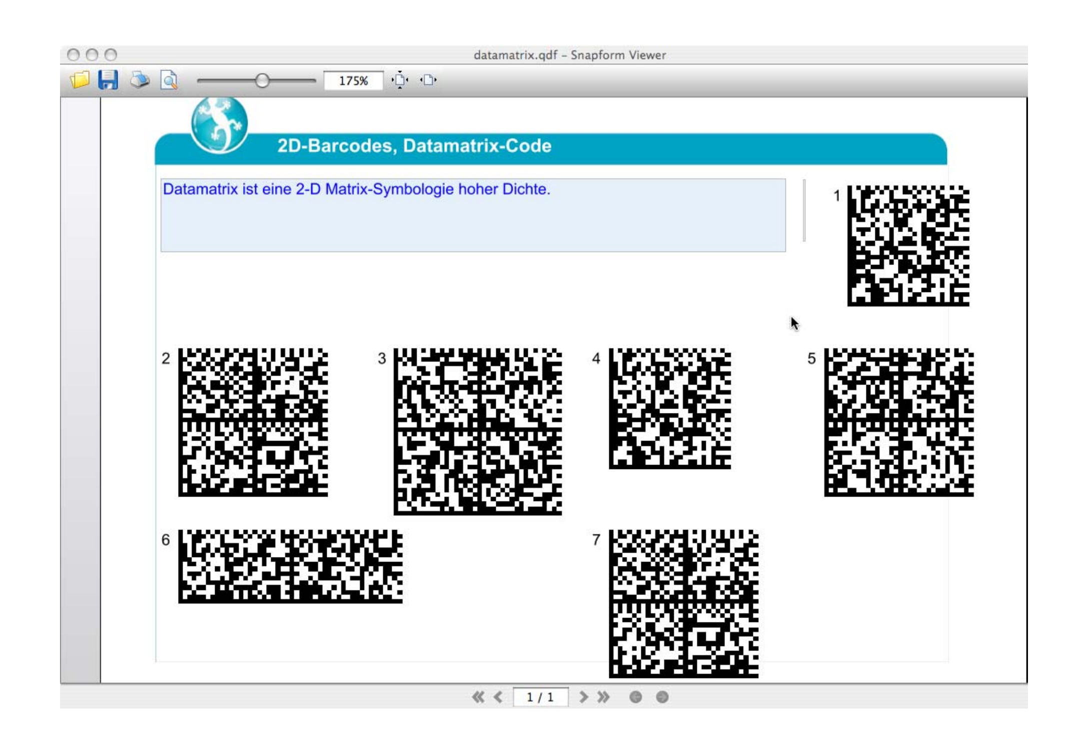


Fig. 4-126 *Beispiele für Datamatrix Barcodes von datamatrix.qdf in Snapform Viewer 085*

1 Bcd000: Datamatrix Barcode, Format und Codierung automatisch
2 Bcd001: Datamatrix Barcode, Format automatisch, Codierung ASCII
3 Bcd002: Datamatrix Barcode, Format automatisch, Codierung C40
4 Bcd003: Datamatrix Barcode, Format automatisch, Codierung TEXT
5 Bcd004: Datamatrix Barcode, Format automatisch, Codierung BASE256
6 Bcd005: Datamatrix Barcode, Format 16x48, Codierung TEXT
7 Bcd006: Datamatrix Barcode, Format 16x48, Codierung ASCII

Alle sieben Barcodes stellen den im Eingabefeld definierten Text entsprechend den gewählten Eigenschaften dar. In der Beispieldatei kann der Text verändert werden, und die Symbole verändern sich entsprechend.

Bei Position 7 sollte ein rechteckiges Symbol erscheinen, aber die gewählte Codierung erzeugt mehr Zeichen als zulässig. Daher wird ein quadratisches Symbol erzeugt. Wird der Eingabetext zu **Datamatrix ist eine 2-D Matrix-Symbologie.** verändert, erscheint das rechteckige Symbol.

4.4.6.5 QRCode

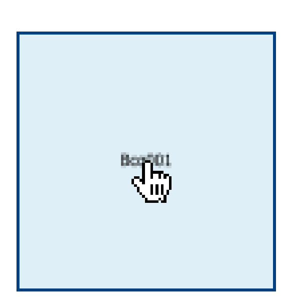
Der QRCode Barcode ist eine quadratische Matrix-Symbologie hoher Dichte. Die Datenkapazität beträgt 7089 numerische Zeichen, 4296 alphanumerische Zeichen, 1817 Kanji-Zeichen oder 2953 Bytes Binärdaten (grösstmöglichstes Symbol, geringste Fehlerkorrektur-Stufe). Der QRCode wurde für Anwendungen in Japan entwickelt. Das Symbol besitzt ein typisches “Findermuster” (ein grösseres Symbol in der linken oberen, rechten oberen und linken unteren Ecke sowie ein kleineres Symbol in der Nähe der unteren rechten Ecke), was eine Bilderfassung aus allen Richtungen erlaubt.

Das Symbol kann eine von 40 definierten Grössen umfassen, und es sind 4 verschiedene Fehlerkorrektur-Stufen möglich.

Neben Zeichen mit Dezimalcode zwischen 0 und 255 werden auch Kanji-Zeichen unterstützt (Hex-Werte 8140 - 9FFC und E040 - EBBF).

Das QRCode Barcode-Werkzeug erzeugt ein Feld für die Anzeige eines QRCode 2D-Barcodes. Nach Wahl des Werkzeugs erscheint ein Cursor mit einem 2D-Barcodefeld in Default-Grösse (siehe Fig. 4-127).

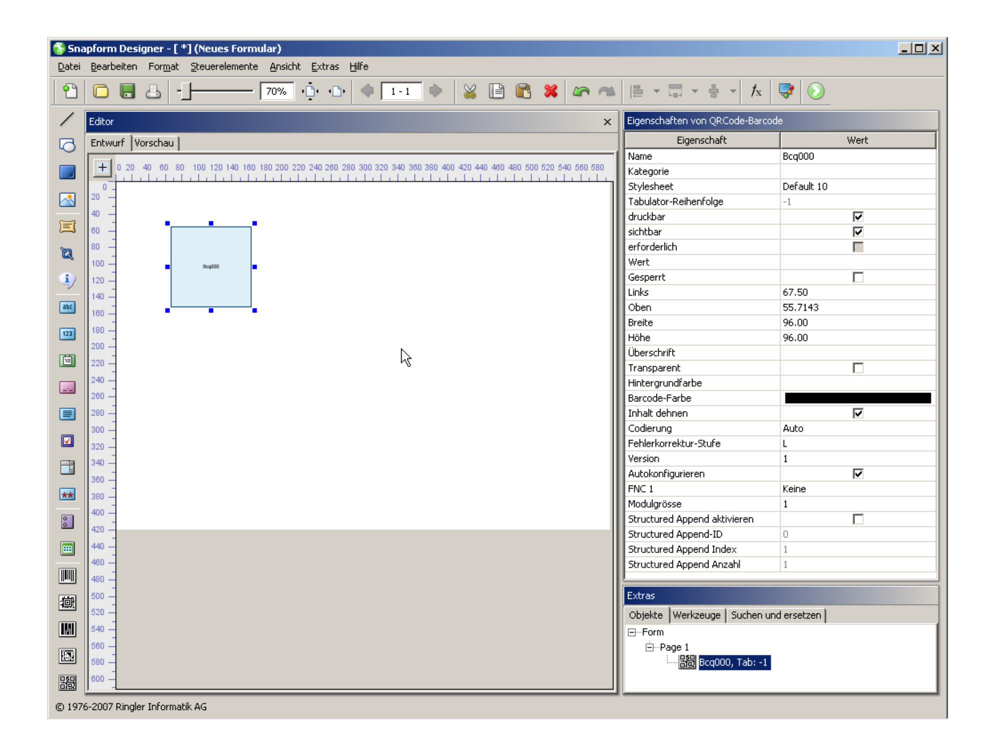
Fig. 4-127 *QRCode Barcode-Werkzeug-Cursor*



Mit dem QRCode Barcode-Werkzeug verknüpft ist ein 2D-Barcodefeld für QRCode, 96 pt breit und 96 pt hoch. Mit Mausklick wird dieses Feld auf der Arbeitsfläche platziert. Dabei erhält es automatisch einen Namen, bestehend aus den Zeichen Bcq und einer dreistelligen fortlaufenden Nummer, beginnend mit 000. Das erste in einem Formular mit dem QRCode Barcode-Werkzeug platzierte Element trägt daher den Namen Bcq000. In der Objektstruktur wird es der betreffenden Seite

hinzugefügt. In Fig. 4-128 ist das erste QRCode Barcode-Feld in einem neuen Formular platziert und anschliessend mit Anklicken selektiert worden.

Fig. 4-128 *Neu erstelltes QRCode Barcode-Feld 086*



Die Länge und Breite des QRCode Barcode-Felds kann durch Ziehen der Ankerpunkte des Selektionsrechtecks verändert werden, die Position auch durch Anfassen und Ziehen. Beim Überfahren der Ankerpunkte ändert der Cursor in Doppelpfeile, welche die Bewegungsrichtung des Ankerpunktes angeben. Beim "Anfassen" und Drücken der Maustaste ändert sich der Cursor zu einem Hohlpfeil.

Die übrigen Merkmale eines QRCode Barcode-Feldes werden über das Eigenschaften-Fenster kontrolliert (siehe Fig. 4-129).

Fig. 4-129 *Eigenschaften eines QRCode Barcodefelds 086*

Eigenschaften von QRCode-Barcode

Eigenschaft	Wert
Name	Bcq000
Kategorie	
Stylesheet	Default 10
Tabulator-Reihenfolge	-1
druckbar	☑
sichtbar	☑
erforderlich	☐
Wert	
Gesperrt	☐
Links	67.50
Oben	55.7143
Breite	96.00
Höhe	96.00
Überschrift	
Transparent	☐
Hintergrundfarbe	
Barcode-Farbe	
Inhalt dehnen	☑
Codierung	Auto
Fehlerkorrektur-Stufe	L
Version	1
Autokonfigurieren	☑
FNC 1	Keine
Modulgrösse	1
Structured Append aktivieren	☐
Structured Append-ID	0
Structured Append Index	1
Structured Append Anzahl	1

Name

Der Name des QRCode Barcodefelds im Dokument. Dieser Name kann geändert werden.

Kategorie

Zugehörigkeit zu einer oder mehreren Kategorien, welche es erlauben, logisch zusammengehörende Gruppen von Elementen zu bilden. Beim Anklicken dieses Felds öffnet sich der Kategorien Dialog (siehe Fig. 4-23).

Die Kategorien werden durch Anklicken des betreffenden Kontrollkästchens ausgewählt. Mit **OK** wird die Auswahl übernommen.

Stylesheet

Das für das Element gültige Stylesheet. Wenn explizite kein Stylesheet ausgewählt worden ist, wird das erste Default-Stylesheet genutzt.

Tabulator-Reihenfolge

Ordnungsnummer des Elements in der Tabulator-Reihenfolge. Dieser Wert ist immer -1 und kann nicht verändert werden. Das bedeutet, dass das Element nicht Bestandteil der Tabulatoren-Reihenfolge ist, was an sich logisch ist, da ein Barcode nicht ein interaktives Element ist.

druckbar

Wenn das Kontrollkästchen gewählt ist, wird das QRCode Barcodefeld ausgedruckt.

sichtbar

Wenn das Kontrollkästchen gewählt ist, wird das QRCode Barcodefeld auf dem Bildschirm angezeigt.

erforderlich

Diese Eigenschaft ist bei QRCode Barcodefeldern immer ausgeschaltet, und kann nicht geändert werden.

Wert

In dieser Eigenschaft des QRCode Barcodefels wird der zu codierende Text eingegeben. Mit Anklicken dieser Eigenschaft öffnet sich der Formeleditor zur Eingabe. In den allermeisten Fällen ist der Wert das Resultat einer Berechnung.

Gesperrt

Wenn das Kontrollkästchen gewählt ist, wird das QRCode Barcodefeld im Snapform Designer blockiert, und kann nicht (versehentlich) verschoben werden. Zudem werden sämtliche übrigen Eigenschaften im Eigenschaften-Dialog deaktiviert. Der Zugriff auf die Eigenschaften ist erst wieder möglich, wenn dieses Kontrollkästchen deaktiviert wird.

Links

X-Koordinate der linken oberen Ecke der Umrandungsbox des Elements in Points. Diese Eigenschaft erlaubt eine genaue numerische Positionierung des QRCode Barcodefelds.

Oben

Y-Koordinate der linken oberen Ecke der Umrandungsbox des Elements in Points. Diese Eigenschaft erlaubt eine genaue numerische Positionierung des QRCode Barcodefelds.

Breite

Breite der Umrandungsbox des QRCode Barcodefelds.

Höhe

Höhe der Umrandungsbox des QRCode Barcodefelds.

Überschrift

Dies ist eine Textzeile, welche oberhalb des QRCode Barcodefelds erscheint. Diese Überschrift kann beispielsweise Informationen für den Sachbearbeiter enthalten, welcher den Code manuell einliest.

Die Überschrift kann entweder als einfacher Text, oder als HTML-Text eingegeben werden. Sie ist von fixer Schriftgrösse (10 pt) und ihre Länge ist auf die Breite des Aztec Barcodefelds beschränkt.

Transparent

Der Hintergrund des QRCode Barcodefelds wird mit diesem Kontrollkästchen auf transparent (keine deckende Hintergrundsfarbe) geschaltet. Diese Eigenschaft ist in der Grundeinstellung ausgeschaltet.

Hintergrundfarbe

Farbe des Barcodefeldes. Mit Anklicken der Eigenschaft öffnet sich das Farbauswahlfenster (siehe Abschnitt 3.2.9.2), welches die Definition der gewünschten Farbe erlaubt.

Die Hintergrundfarbe wird nur dann berücksichtigt, wenn das Kontrollkästchen **Transparent** ausgeschaltet ist.

Barcodefarbe

Farbe des Barcodes. Mit Anklicken der Eigenschaft öffnet sich das Farbauswahlfenster (siehe Abschnitt 3.2.9.2), welches die Definition der gewünschten Farbe erlaubt.

Hinweis: *Die Normen für den QRCode Barcode können Vorschriften bezüglich der zu verwendenden Farben enthalten, und sollten konsultiert werden. Ebenso ist zu beachten, dass einzig die Vollfarben Schwarz, Cyan, Magenta und Yellow vollflächig gedruckt werden (es sei denn, es würden Schmuckfarben eingesetzt), und dass eine gute Druckqualität nur mit diesen Farben bei 100% Sättigung zu nicht aufgerasterten Modulen führt.*

Inhalt dehnen

Dieses Kontrollkästchen bestimmt, ob das Symbol auf die Grösse des QRCode Barcodefelds skaliert wird. Wenn es eingeschaltet ist, wird das Symbol immer auf die Grösse des Felds gebracht. Dabei wird der Wert für die Eigenschaft **Modulgrösse** automatisch übersteuert. Wenn es

ausgeschaltet ist, wird das Symbol mit der festgelegten Modulgrösse erzeugt und in der linken oberen Ecke des Felds positioniert.

Vorsicht: *Wenn diese Eigenschaft eingeschaltet ist, kann bei grösseren Datenmengen das Symbol derart verkleinert werden, dass es nicht korrekt ausgedruckt werden kann, und dass die resultierende Modulgrösse zu klein wird.*

Hinweis: *Wenn diese Eigenschaft ausgeschaltet ist, kann das Symbol grösser als das Feld werden, und eventuell andere Elemente des Dokuments überdecken.*

Codierung

Diese Eigenschaft bestimmt, auf welche Art der im Symbol darzustellende Datenstrom codiert wird. Wenn sie gewählt wird, öffnet sich ein Drop Down Menu mit den Optionen **Auto**, **Byte**, **Numerisch**, **Alphanumerisch** und **Kanji**. **Auto** ist der Grundwert. Mit der Einstellung **Auto** wir die für den zu codierenden Datenstrom passende Codierung benutzt. Die Optionen haben folgende Bedeutung:

Byte

Die Codierung erfolgt Byte-weise. Damit diese Codierung ablaufen kann, muss der Datenstrom als Bytefolge vorliegen. Dies geschieht mit Hilfe der Funktionen getByte(string) bzw. Compress(x).

Numerisch

Die Codierung erfolgt nur für Zahlen 0 - 9.

Alphanumerisch

Die Codierung erfolgt nur für Zahlen 0-9, Grossbuchstaben A - Z sowie 9 Sonderzeichen <Space> $ % * + - . / :

Kanji

Die Codierung erfolgt nur für japanische Zeichen, mit Hex Code 8140 - 9FFC und E040 - EBBF. In diesen Bereichen sind Zahlen und Grossbuchstaben sowie gewisse Sonderzeichen ebenfalls enthalten.

Wenn der Datenstrom Zeichen enthält, welche nicht codiert werden können, erscheint das QRCode Barcodefeld als graue Fläche.

Fehlerkorrektur-Stufe

Der QRCode Barcode kennt verschiedene Stufen der Fehlerkorrektur. Die geringste Stufe erlaubt einen Verlust von etwa 7% des Symbols, die höchste etwa 30%.

Wenn diese Eigenschaft ausgewählt wird, öffnet sich ein Drop Down Menu mit den Optionen **L**, **M**, **Q** und **H**. Der Grundwert ist **L**.

Version

Im Sprachgebrauch des QRCode Barcodes steht der Begriff "Version" für die Grösse des Symbols. Es sind 40 Grössen definiert, welche für die Codierung in Frage kommen.

Wenn diese Eigenschaft ausgewählt wird, öffnet sich ein Drop Down Menu mit den Optionen entsprechend den 40 verschiedenen Grössen (Werte zwischen **1** und **40**). Der Grundwert ist **1**.

Wenn die Eigenschaft **Autokonfigurieren** eingeschaltet ist, wird bei zunehmender Datenmenge das nächstgrössere Symbol gewählt. Wenn sie ausgeschaltet ist, wird das QRCode Barcodefeld bei zu grosser Datenmenge als graue Fläche dargestellt.

Autokonfigurieren

Wenn dieses Kontrollkästchen gewählt ist, wird für das QRCode Barcodesymbol bei zunehmender Datenmenge automatisch die nächstgrössere Versionsnummer gewählt. Das Symbol wächst entsprechend (oder die Modulgrösse verringert sich, falls Dehnen eingeschaltet ist). Grundstellung dieser Eigenschaft ist ausgeschaltet.

FNC 1

Der QRCode Barcode kennt ein spezielles Steuerzeichen FNC1, welches für bestimmte normierte Zwecke eingesetzt werden kann. Es steht entweder an erster oder zweiter Stelle des Datenstroms.

Wenn diese Eigenschaft ausgewählt wird, öffnet sich ein Drop Down Menu mit den Optionen **Kein**, **1** und **2**. Der Grundwert ist **Kein**.

Die Option **Kein** bedeutet, dass kein FNC1-Zeichen verwendet wird. Die Option **1** bedeutet, dass das FNC1-Zeichen an erstere Stelle im Datenstrom steht, und besagt, dass die codierten Daten gemäss den

UCC/EAN Application Identifiers-Normen formatiert sind. Die Option **2** bedeutet, dass das FNC1-Zeichen an zweiter Stelle im Datenstrom steht, und besagt, dass die codierten Daten gemäss separat spezifizierten, bei AIM International registrierten, Industrie-Normen formatiert sind.

Modulgrösse

Dies ist das Mass eines Moduls (der kleinsten Einheit des QRCode Barcodes). Die Masseinheit ist Point, und der Grundwert ist 1. Der Wert sollte nicht kleiner sein, da sonst Probleme beim Druck entstehen könnten.

Hinweis: *Wenn die Eigenschart Dehnen gewählt ist, wird der Wert für die Modulgrösse übersteuert.*

Structured Append aktivieren

Der QRCode Barcode erlaubt es, bis zu 16 Symbole miteinander zu verketten, und dadurch eine entsprechend höhere Datenkapazität zu erreichen. Beim Erfassen verketteter Symbole speichert die Leseeinheit die Daten und fügt sie zusammen. Erst wenn alle Symbole eingelesen sind, werden die Daten ausgegeben. Gegenüber der nachgeschalteten Applikation erscheint das Ganze, als ob ein riesiges Symbol eingelesen worden wäre.

Wenn dieses Kontrollkästchen gewählt ist, ist das betreffende Symbol Bestandteil einer strukturell verketteten Gruppe. Im anderen Fall ist das Symbol eigenständig.

Wenn diese Eigenschaft gewählt ist, müssen die Eigenschaften **Structured Append Anzahl** und **Structured Append Index** einen Wert aufweisen; sie werden im Eigenschaftenfenster auch aktiviert. Die Eigenschaft Structured Append-ID wird ebenfalls aktiviert, ist jedoch nicht zwingend mit einem Wert zu versehen.

Structured Append Anzahl

Diese Eigenschaft gibt an, wieviele Symbole in der strukturell verketteten Gruppe vorhanden sind, deren Bestandteil das betreffende Symbol ist.

Gültige Werte liegen zwischen **2** und **16**. Der Wert **1** bedeutet, dass die Gruppe aus einen Symbol besteht, und das Symbol daher eigenständig ist.

Diese Eigenschaft ist normalerweise deaktiviert, wird aktiviert, wenn das Kontrollkästchen **Structured Append aktivieren** eingeschaltet wird.

Structured Append Index

Diese Eigenschaft ist die Ordnungsnummer des Symbols in der strukturell verketteten Gruppe.

Gültige Werte liegen zwischen **1** und **16**.

Diese Eigenschaft ist normalerweise deaktiviert. Sie wird aktiviert, wenn das Kontrollkästchen **Structured Append aktivieren** eingeschaltet wird.

Structured Append-ID

Es ist möglich, den Dateinamen für die ausgelesenen verketteten Daten im voraus zu definieren. Dieser Dateiname wird in dieser Eigenschaft festgelegt.

Diese Eigenschaft ist normalerweise deaktiviert. Sie wird aktiviert wenn das Kontrollkästchen **Structured Append aktivieren** eingeschaltet wird. Sie ist jedoch optional und wird daher nur applikationsbedingt benötigt.

Im Dokument qrcode.qdf (siehe Fig. 4-115 (Entwurfsmodus)) bzw. Fig. 4-116 (Ansicht in Snapform Viewer)) sind verschiedene Beispiele von QRCode Barcodes gezeigt.

Nach dem Selektieren der Barcodefelder im Snapform Designer gibt das Eigenschaften-Fenster Auskunft über deren jeweiligen Merkmale. Ein Doppelklick auf das Feld öffnet den Formeleditor zur Anzeige der hinterlegten Formel.

Fig. 4-130 *Beispiele für QRCode Barcodes von qrcode.qdf, im Entwurfs-Modus 087*

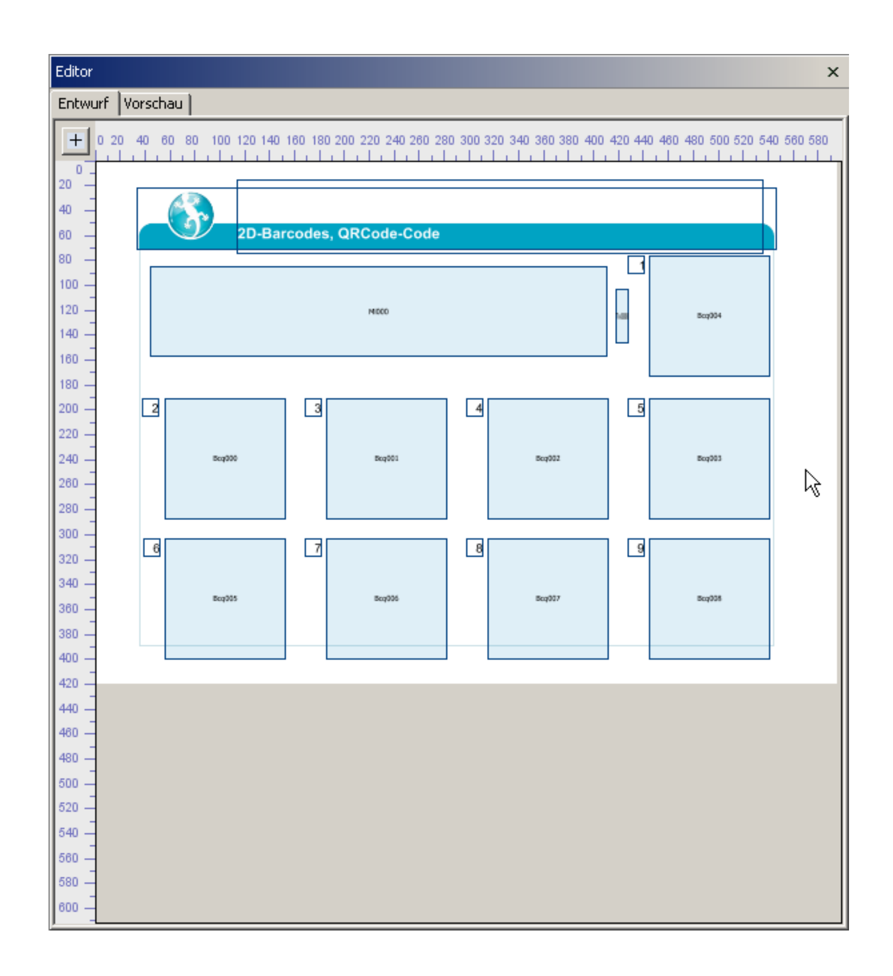
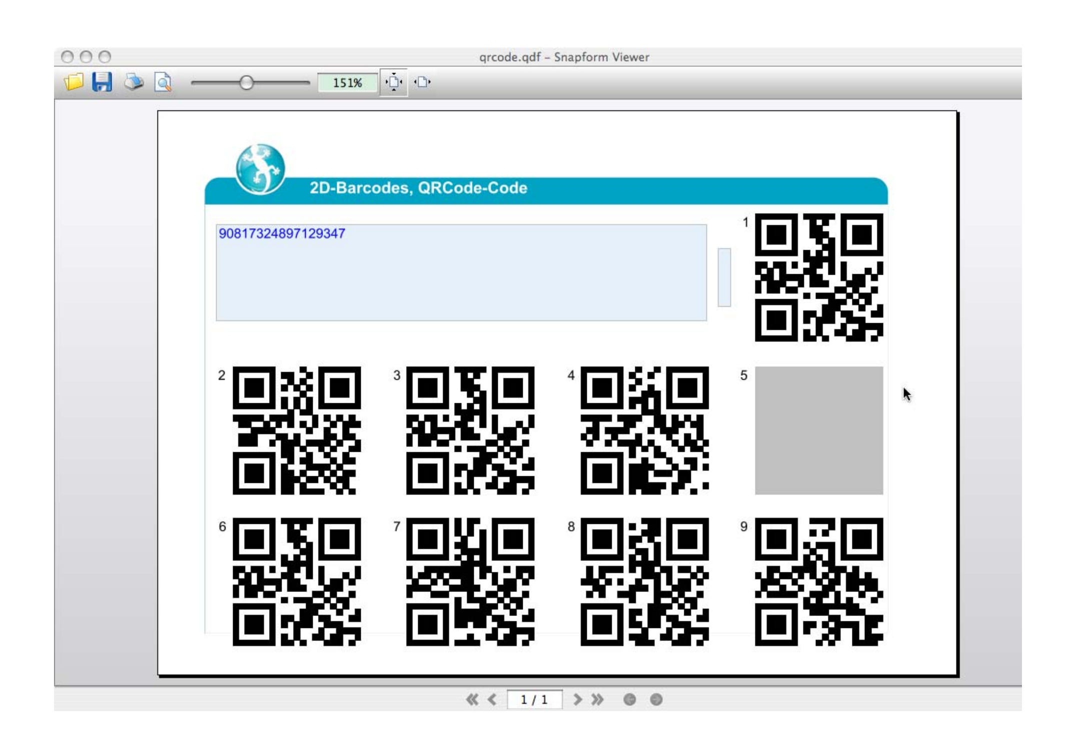


Fig. 4-131 *Beispiele für QRCode Barcodes von qrcode.qdf in Snapform Viewer 088*



1 Bcq004: QRCode Barcode, Codierung automatisch
2 Bcq000: QRCode Barcode, Codierung Binär; für die Berechnung des Datenstroms wurde die Funktion **getBytes(MI000)** verwendet
3 Bcq001: QRCode Barcode, Codierung Numerisch
4 Bcq002: QRCode Barcode, Codierung Alphanumerisch
5 Bcq003: QRCode Barcode, Codierung Kanji
6 Bcq005: QRCode Barcode, Codierung automatisch, Fehlerkorrekturstufe L
7 Bcq006: QRCode Barcode, Codierung automatisch, Fehlerkorrekturstufe M
8 Bcq007: QRCode Barcode, Codierung automatisch, Fehlerkorrekturstufe Q
9 Bcq008: QRCode Barcode, Codierung automatisch, Fehlerkorrekturstufe H

Alle Barcodes stellen den im Eingabefeld definierten Text entsprechend den gewählten Eigenschaften dar. In der Beispieldatei kann der Text verändert werden, und die Symbole verändern sich entsprechend.

Das Symbol mit Position 5 (Codierung Kanji) setzt Eingaben betreffenden Zeichenbereich voraus.

4.4.7 Tabellen

In vielen Formularen werden Daten in tabellarischer Form dargestellt. Häufig ist es dann auch so, dass die Anzahl Zeilen der Tabellen nicht vorhergesagt werden kann. Bei Papier-Formularen muss ein optimaler Wert angenommen werden (welcher sich meistens auch nach dem verfügbaren Platz richtet). Wenn die Tabelle dann mehr Zeilen erhält, muss mit irgendwelchen Beiblättern gearbeitet werden.

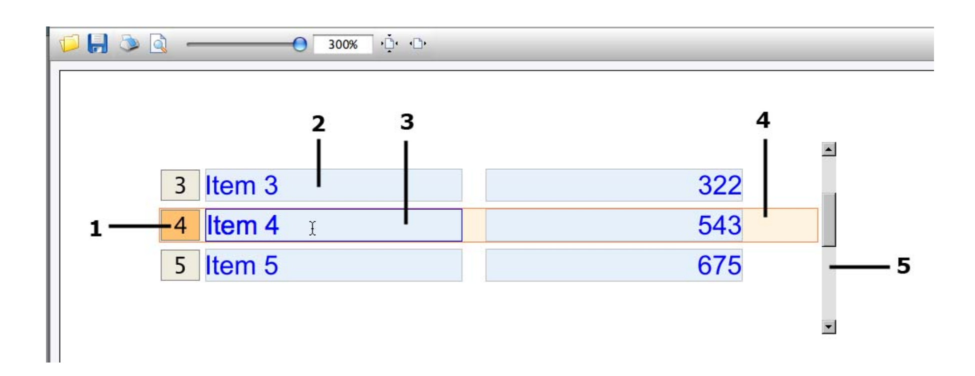
Bei elektronischen Formularen gibt es die Möglichkeit, Tabellen wachsen zu lassen, und die weiteren Formularelemente “nach unten zu schieben”. Dies hat bei Bildschirmanzeigen eine gute Berechtigung, da meist sowieso gescrollt werden muss. Das Problem taucht aber spätestens dann auf, wenn das Formular doch noch gedruckt werden muss. Hier ist die Seitenaufteilung integraler Bestandteil des Formulars, und darf nicht einfach so verändert werden.

Bei Snapform wird ein grosses Gewicht auf die Seitentreue gelegt. Daher wird für Tabellen mit Datenüberlauf ein Beiblatt erzeugt, und im eigentlichen Formular können zusammenfassende Informationen angezeigt werden.

Die Grundlage für Tabellen in Snapform ist ein Tabellencontainer, welchem die einzelnen Tabellenelemente zugeordnet werden. Das Verhalten der Tabelle wird über die Eigenschaftendes Tabellencontainers bestimmt.

Tabellen haben in Snapform-Formularen verschiedene Bildelemente, welche in Fig. 4-132 dargestellt sind.

Fig. 4-132 *Elemente einer Tabelle in Snapform*



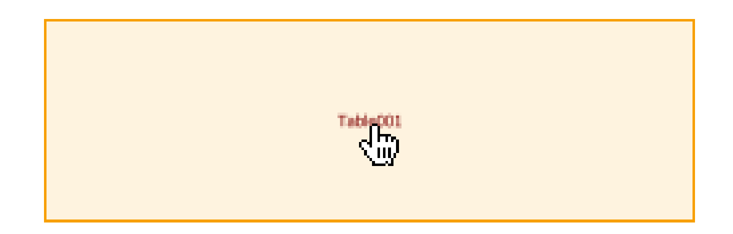
1 Selektor: Dieses Element dient zum Selektieren der ganzen Tabellenzeile. Er ist nummeriert, wobei der Startwert eine Tabelleneigenschaft ist.
2 Feld in der Tabelle: Innerhalb der Tabellenfelder kann auch mit den Cursorsteuertasten navigiert werden.
3 Aktives Feld: Dieses Feld ist aktiv.
4 Aktive Zeile: Diese Zeile ist durch die Aktivierung des Felds (Position 3) ebenfalls aktiviert worden.
5 Rollbalken: Wenn mehr Tabellenzeilen vorhanden sind, als angezeigt werden können, erscheint der Rollbalken, und erlaubt ein Navigieren innerhalb der Tabelle.

4.4.7.1 Das Tabellenwerkzeug

Der Tabellencontainer ist ein Feld, welches jedes neu erzeugte und darin platzierte Eingabefeld zu einem Bestandteil der Tabellen-Gruppe macht. In der Objektstruktur sind diese Felder eine Ebene tiefer als die übrigen Felder. Der Tabellencontainer ist ausserhalb der Entwurfs-Ansicht auch unsichtbar (Vorschau und Snapform Viewer).

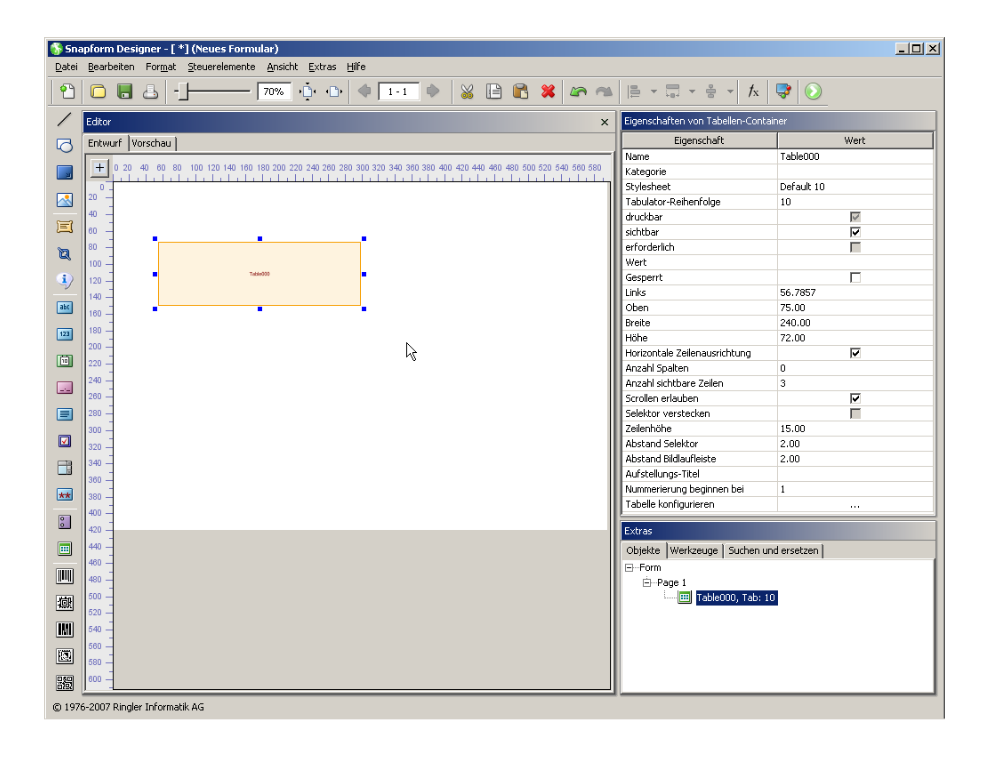
Das Tabellenwerkzeug erzeugt einen Tabellencontainer. Nach Wahl des Werkzeugs erscheint ein Cursor mit einem Tabellencontainer in Default-Grösse (siehe Fig. 4-133).

Fig. 4-133 *Tabellencontainer-Cursor*



Mit dem Tabellenwerkzeug verknüpft ist ein Tabellencontainer, 240 pt breit und 72 pt hoch. Mit Mausklick wird dieses Feld auf der Arbeitsfläche platziert. Dabei erhält es automatisch einen Namen, bestehend aus den Zeichen Table und einer dreistelligen fortlaufenden Nummer, beginnend mit 000. Der erste in einem Formular mit dem Tabellenwerkzeug platzierte Container trägt daher den Namen Table000. In der Objektstruktur wird er der betreffenden Seite hinzugefügt. In Fig. 4-134 ist der erste Tabellencontainer in einem neuen Formular platziert, und anschliessend mit Anklicken selektiert worden.

Fig. 4-134 *Neu erstellter Tabellencontainer 089*



Die Länge und Breite des Tabellencontainers kann durch Ziehen der Ankerpunkte des Selektionsrechtecks verändert werden, die Position

auch durch Anfassen und Ziehen. Beim Überfahren der Ankerpunkte ändert der Cursor in Doppelpfeile, welche die Bewegungsrichtung des Ankerpunktes angeben. Beim “Anfassen” und Drücken der Maustaste ändert sich der Cursor zu einem Hohlpfeil.

Der nächste Schritt zum Erstellen ist das Hinzufügen von Tabellenelementen (siehe Abschnitt 4.4.7.2).

Die übrigen Merkmale eines Tabellencontainers werden über das Eigenschaften-Fenster kontrolliert (siehe Fig. 4-135).

Fig. 4-135 *Eigenschaften eines Tabellencontainers 089*

Eigenschaften von Tabellen-Container

Eigenschaft	Wert
Name	Table000
Kategorie	
Stylesheet	Default 10
Tabulator-Reihenfolge	10
druckbar	☑
sichtbar	☑
erforderlich	☐
Wert	
Gesperrt	☐
Links	56.7857
Oben	75.00
Breite	240.00
Höhe	72.00
Horizontale Zeilenausrichtung	☑
Anzahl Spalten	0
Anzahl sichtbare Zeilen	3
Scrollen erlauben	☑
Selektor verstecken	☐
Zeilenhöhe	15.00
Abstand Selektor	2.00
Abstand Bildlaufleiste	2.00
Aufstellungs-Titel	
Nummerierung beginnen bei	1
Tabelle konfigurieren	...

gDer Name des Tabellencontainers im Dokument. Dieser Name kann geändert werden.

Kategorie

Zugehörigkeit zu einer oder mehreren Kategorien, welche es erlauben, logisch zusammengehörende Gruppen von Elementen zu bilden. Beim Anklicken dieses Felds öffnet sich der Kategorien Dialog (siehe Fig. 4-23).

Die Kategorien werden durch Anklicken des betreffenden Kontrollkästchens ausgewählt. Mit **OK** wird die Auswahl übernommen.

Stylesheet

Das für das Element gültige Stylesheet. Wenn explizite kein Stylesheet ausgewählt worden ist, wird das erste Default-Stylesheet genutzt. Bei Tabellen sind insbesondere die unter dem Reiter Tabellen Farben definierten Eigenschaften von Bedeutung.

Tabulator-Reihenfolge

Ordnungsnummer des Elements in der Tabulator-Reihenfolge. Dieser Wert wird automatisch vergeben, und liegt in der Reihe der Feld-Tabulator-Reihenfolge (welche mit 10 beginnend in Zehnerschritten erhöht wird). Das erste Feld des Formulars hat daher den Tabulator-Reihenfolge-Wert 10, das zweite den Wert 20, etc.. Der Wert kann verändert werden, so dass eine frei bestimmbare Tabulatorreihenfolge aufgebaut werden kann, welche nicht mit der Reihenfolge des Hinzufügens der Felder übereinstimmen muss.

Bei Tabellen ist darauf zu achten, dass der Tabellencontainer die niedrigste Ordnungsnummer der ganzen Tabellengruppe aufweist.

druckbar

Dieses Kontrollkästchen ist deaktiviert, und kann nicht verändert werden. Tabellencontainer werden nie ausgedruckt.

sichtbar

Wenn dieses Kontrollkästchen gewählt ist, wird der Tabellencontainer einschliesslich dessen zugeordneten Feldelemente auf dem Bildschirm angezeigt.

erforderlich

Diese Eigenschaft ist bei Tabellencontainern immer ausgeschaltet, und kann nicht geändert werden.

Wert

Diese Eigenschaft ist bei Tabellencontainern zwar vorhanden, aber nicht aktiv, und kann auch nicht aktiviert werden.

Gesperrt

Wenn dieses Kontrollkästchen gewählt ist, wird der Tabellencontainer einschliesslich der darin enthaltenen Formularelemente im Snapform Designer blockiert, und kann nicht (versehentlich) verschoben werden. Zudem werden sämtliche übrigen Eigenschaften im Eigenschaften-

Dialog deaktiviert (ebenso für die enthaltenen Kontrollkästchen-Felder). Der Zugriff auf die Eigenschaften ist erst wieder möglich, wenn dieses Kontrollkästchen deaktiviert wird.

Die zum Container gehörenden Formularelemente können zwar selektiert und deren Eigenschaften angezeigt werden, aber ein Verändern ist nicht möglich.

Links

X-Koordinate der linken oberen Ecke der Umrandungsbox des Elements in Points. Diese Eigenschaft erlaubt eine genaue numerische Positionierung des Tabellencontainers.

Oben

Y-Koordinate der linken oberen Ecke der Umrandungsbox des Elements in Points. Diese Eigenschaft erlaubt eine genaue numerische Positionierung des Tabellencontainers.

Breite

Breite der Umrandungsbox des Tabellencontainers.

Höhe

Höhe der Umrandungsbox des Tabellencontainers.

Horizontale Zeilenausrichtung

Diese Eigenschaft bestimmt, in welche Richtung eine “logische Zeile” verläuft. Eine “logische Zeile” entspricht der in der Tabelle aufgeführten logischen Gruppe. Normalerweise ist die logische Zeile horizontal, sie kann aber auch vertikal sein (beispielsweise in einem klassischen Fahrplanfeld).

Hinweis: *Eine “visuelle Zeile” ist hingegen, wie die Bezeichnung sagt, rein visuell, und steht nicht in einem zwingenden Zusammenhang mit den in der Tabelle dargestellten Informationen. Eine visuelle Zeile verläuft immer horizontal, eine visuelle Spalte vertikal. In den nachfolgenden Beschreibungen wird jeweils explizite auf logische oder visuelle Zeile bzw. Spalte Bezug genommen.*

Wenn dieses Kontrollkästchen gewählt ist, verläuft eine logische Zeile horizontal, und zusätzliche Zeilen werden unterhalb der letzten

aktuellen Zeile angefügt. Im anderen Fall verläuft eine logische Zeile vertikal, und zusätzliche Zeilen werden rechts der letzten aktuellen Zeile angefügt. Der Grundwert dieser Eigenschaft ist eingeschaltet.

Hinweis: *Wenn diese Eigenschaft umgeschaltet wird, muss beachtet werden, dass verschiedene weitere Eigenschaften "gedreht" sind. Es ist insbesondere darauf zu achten, dass der Wert der Eigenschaft* **Zeilenhöhe** *nun die visuelle Spaltenbreite darstellt, und entsprechend angepasst werden muss.*

Anzahl Spalten

Diese Eigenschaft gibt an, wieviele Spalten in der Tabelle definiert sind. Dieser Wert entspricht im Wesentlichen der Anzahl Formularelemente, welche sich im Container befinden (Checkboxen, welche durch einen Radiobutton-Container zusammengefasst worden sind, zählen einzeln). Mit dem Hinzufügen von weiteren Elementen zur Tabelle ändert sich dieser Wert automatisch.

Dieser Wert dient zur Information. er kann zwar verändert werden, die Änderung wird jedoch verworfen.

Anzahl sichtbare Zeilen

Diese Eigenschaft gibt an, wieviele logische Zeilen der Tabelle jeweils angezeigt werden. Der Grundwert ist 3.

Die tatsächlich angezeigte Anzahl Zeilen hängt vom Wert der Eigenschaft **Zeilenhöhe** und den Massen des Tabellencontainers ab. Wenn der Tabellencontainer zuwenig gross ist, werden nur soviele Zeilen angezeigt, wie im Perimeter des Tabellencontainers Platz haben. Zeilen können angeschnitten dargestellt werden.

Ein Spezialfall ist, wenn dieser Wert auf **1** gesetzt wird. Dies führt zu einer sogenannten "Phantom-Tabelle". Da nur eine Zeile zur Verfügung steht, können Eingaben nicht richtig erfolgen, und auch die Anzeige der Informationen ist sehr eingeschränkt. Um trotzdem die Daten anzeigen zu können wird für diese Art Tabellen eine "Aufstellung" genannte Hilfstabelle benutzt, welche alle Daten der Tabelle zugänglich macht. Diese Tabelle wird über die Eigenschaft **Tabelle konfigurieren** definiert.

Scrollen erlauben

Snapform erlaubt es, Tabellen mit nicht vorbestimmter Anzahl logischer Zeilen aufzubauen. Diese Eigenschaft bestimmt, ob über die in **Anzahl sichtbare Zeilen** festgelegte Anzahl hinausgehende Zeilen hinzugefügt werden können. Wenn die effektive Anzahl logischer Zeilen die Anzahl sichtbarer Zeilen überschreitet, erscheint automatisch ein Rollbalken, welcher ein Rollen der angezeigten Zeilen erlaubt.

Wenn dieses Kontrollkästchen gewählt ist, können einer Tabelle beliebig viele zusätzliche Zeilen hinzugefügt werden. Im anderen Fall ist die Anzahl Zeilen beschränkt.

Wenn mehr Zeilen vorhanden sind, als angezeigt werden können, werden im Snapform Viewer beim Drucken (oder in der Druckvoransicht) Zusatzblätter erzeugt, welches sämtliche Tabellenwerte in einer einfachen tabellarischen Zusammenstellung zeigen. Diese Tabelle wird “Aufstellung” genannt. Der Aufbau dieser Tabelle wird über die Eigenschaft **Tabelle konfigurieren** kontrolliert.

Zeilenhöhe

Diese Eigenschaft bestimmt die Höhe der logischen Zeile der Tabelle. Dieser Wert setzt den Abstand in Spaltenrichtung zwischen zwei entsprechenden Feldern der logischen Zeilen fest. Im Normalfall ist dies der Abstand zwischen visuellen Zeilen. Die Masseinheit ist Points, und der Grundwert beträgt **15** (was sich sehr gut mit dem Grundwert der Höhe von Eingabefeldern verträgt).

Praxistip: *Damit bei der Anzeige der Tabelle keine Überschneidungen auftreten, muss dieser Wert gegebenenfalls angepasst werden. Als Faustregel kann “Höhe der Umrandung um alle Felder einer Zeile plus 3 Point” gelten. Wenn alle Felder sauber auf einer horizontalen Linie ausgerichtet sind, bedeutet dies auch “Höhe des höchsten Felds in der Zeile plus 3 Point”.*

Hinweis: *Wenn die Eigenschaft Vertikale Zeilenausrichtung ausgeschaltet ist, muss dieser Wert kontrolliert werden. Damit keine Überschneidungen auftreten, ist die im obigen Praxistip erwähnte Faustregel sinngemäss auf die Feldbreite anzuwenden.*

Abstand Selektor

Der Selektor (siehe Fig. 4-132) dient zum Kennzeichnen und Selektieren von logischen Tabellenzeilen. Er wird automatisch links (bzw. oberhalb) der logischen Zeile platziert ausserhalb des Tabellencontainers angezeigt.

Diese Eigenschaft steuert den Abstand des Selektors vom Rand des Tabellencontainers. Die Masseinheit ist Points, und der Grundwert beträgt 2 Pt. Der Selektor hat dieselbe Höhe wie das Feld der ersten Tabellenspalte und ist ungefähr quadratisch, so dass zwei Ziffern ohne Skalieren dargestellt werden können.

Diese Abmessungen müssen bei der Positionierung von Tabellen berücksichtigt werden.

Abstand Bildlaufleiste

Wenn die Eigenschaft **Scrollen erlauben** aktiv ist, und die Tabelle mehr Zeilen als darstellbar enthält, erscheint auf der rechten (bzw. unteren) Seite des Tabellencontainers ein Rollbalken, welcher die Navigation zwischen den Tabellenzeilen erleichtert (siehe Fig. 4-132).

Diese Eigenschaft steuert die Position des Rollbalkens gegenüber dem Tabellencontainer. Die Masseinheit ist Points, und der Grundwert beträgt 2 Pt. Das Bildelemente des Rollbalkens werden durch das Betriebssystem zur Verfügung gestellt, und können durch Snapform nicht beeinflusst werden.

Diese Abmessungen müssen bei der Positionierung von Tabellen berücksichtigt werden. Es empfiehlt sich, in wichtigen Fällen das Formular auf allen zu erwartenden Plattformen zu testen.

Aufstellungs-Titel

Diese Eigenschaft bestimmt den Titel, welcher auf den Aufstellungstabellen ausgedruckt wird. Zudem wird dieser Wert auch in der Titelleiste der Datenerfassungstabelle bei Phantomtabellen angezeigt.

Mit Anklicken dieser Eigenschaft öffnet sich der Formeleditor zur Eingabe. Der Wert kann entweder ein fixer Wert oder das Resultat einer

Berechnung sein. Diese Eigenschaft unterstützt nur einfache Texte; HTML-Tags werden nicht interpretiert.

Nummerierung beginnen bei

Diese Eigenschaft legt den Beginn der Nummerierung der Zeilen in den Selektoren fest (siehe Fig. 4-132). Der Grundwert ist **1**. Es können nur ganze Zahlen eingegeben werden.

Tabelle konfigurieren

Diese Eigenschaft definiert die Aufstellung genannte Tabelle. Diese Tabelle wird einerseits in einer Druckversion angezeigt, wenn Tabelle im Formular mehr Zeilen enthält, als Platz vorhanden ist. Andererseits wird sie bei Phantom-Tabellen auch als Dateneingabetabelle benutzt.

Mit dem Anklicken von **...** in der Wertespalte dieser Eigenschaft öffnet sich das Konfigurationsfenster (siehe Fig. 4-136).

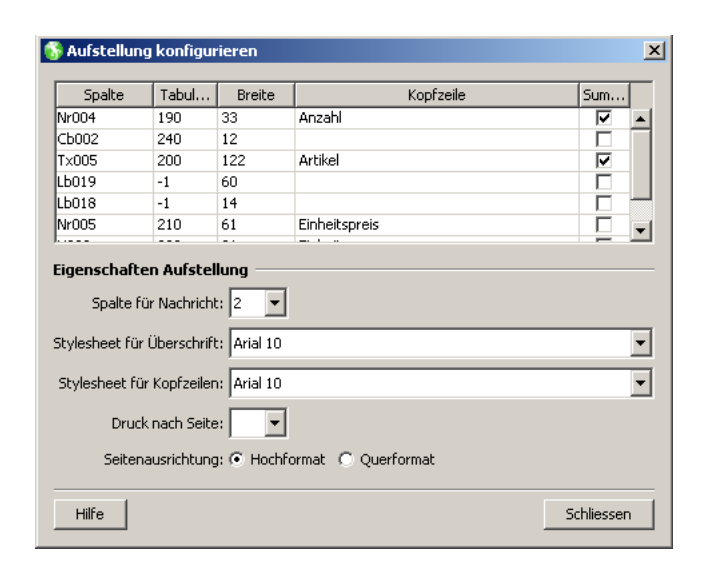


Fig. 4-136 *Konfigurations-Fenster für Aufstellung-Tabelle 090*

Das Konfigurations-Fenster besteht aus zwei Zonen, der Tabellendefinition und den Aufstellung-Eigenschaften. Die Zone der Tabellendefinition ist selbst eine fünfspaltige Tabelle zur Definition der Aufstellung-Tabelle.

Spalte

Bezeichnung des Formularelements der entsprechenden Spalte.

Vorsicht: *Wenn die Bezeichnung der Spalte in dieser Tabelle geändert wird, wird auch der Name des Formularelements geändert, was unter Umständen zu Fehlfunktionen in Formeln führen kann.*

Tabulatorenfolge

Ordnungsnummer des Formularelements der entsprechenden Spalte.

Vorsicht: *Die Ordnungsnummer bestimmt nicht nur die Tabulatoren-Reihenfolge für die Dateneingabetabelle, sondern auch diejenige im Formular selbst. Wenn dieser Wert geändert wird, ändert sich die gesamte Tabulatoren-Reihenfolge des Formulars. Es ist hier leicht möglich, eine Ordnungsnummer doppelt zu belegen.*

Breite

Breite der entsprechenden Spalte in der Aufstellung-Tabelle. Als Ausgangswert wird die Breite des zugehörigen Formularelements genommen. Bei der Darstellung im Aufstellung-Fenster oder in der Druckansicht werden die Spalten proportional auf die Gesamtbreite skaliert. Daher ist es auch zulässig, für diese Eigenschaft den prozentualen Anteil an der Gesamtbreite einzugeben.

Kopfzeile

Dies ist der Titel oberhalb der Spalte. Er erscheint sowohl in der Dateneingabetabelle, wie auch in der Druckansicht. Dieser Text ist frei wählbar und hat kein Äquivalent auf dem Formular selbst.

Summe anzeigen

Dieses Kontrollkästchen hat eine Doppelfunktion. Wenn es gewählt ist, erscheint bei “Phantom-Tabellen” links des betreffenden Felds eine Schaltfläche zum Öffnen des Dateneingabefensters. Zudem wird (zumindest bei Spalten aus Zahlenfeldern) die Summe der Einträge gebildet und auf dem Formular, im Dateineingabefenster und in der Druckansicht angezeigt.

Hinweis: *Damit bei “Phantom-Tabellen” eine Schaltfläche zum Öffnen des Dateneingabefensters angezeigt wird, muss mindestens ein Kontrollkästchen dieser Eigenschaft gewählt sein.*

Unterhalb dieser Tabelle sind verschiedene Eigenschaften für die Aufstellung-Tabelle aufgeführt.

Spalte für Nachricht

In der Druckansicht wird bei Tabellen mit mehr Daten als im Formular angezeigt werden kann die "Aufstellung" erzeugt. In diesem Fall erscheinen im Formular selbst keine Detaildaten mehr, und anstelle dessen wird der **Siehe Aufstellung** angezeigt. Dieser Text erscheint auch bei "Phantom-Tabellen", wenn mehr als eine Zeile Daten vorhanden ist.

Diese Eigenschaft legt fest, in welcher Spalte dieser Text angezeigt werden soll. Das sich beim Anklicken dieses Felds öffnende Drop Down Menu umfasst die Nummern der Formularspalten. In der gewählten Spalte erscheint dann der Hinweistext. Der Hinweistext ist fest programmiert und kann nicht verändert werden.

Hinweis: *Bei "Phantom-Tabellen" sind die folgenden Punkte zu beachten: 1. Das Formularelement für die entsprechende Spalte muss in der Lage sein, Text anzuzeigen. 2. Damit der Hinweistext angezeigt wird, muss für die gewählte Spalte das Kontrollkästchen* **Summe anzeigen** *gewählt worden sein.*

Stylesheet für Überschrift

Diese Eigenschaft legt das Stylesheet für die Überschrift der Aufstellung-Tabelle fest. Im Wesentlichen wird vom gewählten Stylesheet die Schriftart zur Anwendung gebracht. Das sich beim Anklicken der Eigenschaft öffnende Drop Down Menu umfasst die Liste der für das Formular verfügbaren Stylesheets.

Der Text der Überschrift wird in der Tabellencontainer-Eigenschaft **Aufstellungs-Titel** definiert.

Stylesheet für Kopfzeilen

Diese Eigenschaft legt das Stylesheet für die Spaltenüberschriften der Aufstellung-Tabelle fest. Im Wesentlichen wird vom gewählten Stylesheet die Schriftart zur Anwendung gebracht. Das sich beim

Anklicken der Eigenschaft öffnende Drop Down Menu umfasst die Liste der verfügbaren Stylesheets.

Der Text für die Spaltenüberschriften wird in der Spalte **Kopfzeile** der Konfigurationstabelle festgelegt.

Druck nach Seite

Diese Eigenschaft bestimmt, nach welcher Seite des Formulars die Aufstellung-Tabelle beim Ausdruck eingefügt wird.

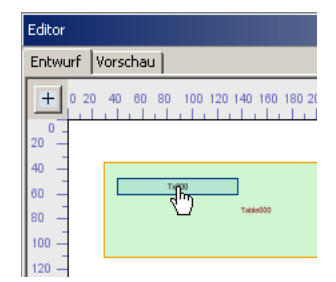
Das sich beim Anklicken dieser Eigenschaften öffnende Drop Down Menu umfasst die Liste der Seitennummern des Formulars. Wenn kein Wert gewählt wird, wird die Aufstellung-Tabelle nach der letzten Seite des Formulars eingefügt.

4.4.7.2 Aufbau von Tabellen

Nachdem der Tabellencontainer platziert worden ist, geht es nun darum, die Tabelle selbst zusammen zu stellen. Dies geschieht dadurch, dass innerhalb des Tabellencontainers eine Musterzeile erzeugt wird, indem die entsprechenden Formularelemente platziert werden. Anschliessend sind gegebenenfalls verschiedene Tabelleneigenschaften anzupassen (siehe Abschnitt 4.4.7.1)

Formularelemente, innerhalb des Tabellencontainers platziert werden, werden diesem zugeordnet, und werden Bestandteil der Tabellenzeile. Wenn ein Formularelement dem Tabellencontainer zugeordnet wird, ändert dieser seine Farbe (siehe Fig. 4-137).

Fig. 4-137 *Formularelemente in Tabellencontainer platzieren 091*

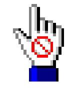


Die Farbänderung ist die visuelle Anzeige, dass das Formularelement angenommen wird.

Die meisten Formularelemente können Bestandteil einer Tabellenzeile werden. Nicht platziert werden können die folgenden Elemente: Linie, Rechteck, Bild, Hinweis/Infopoint sowie Passwortfeld. Bei diesen nicht-platzierbaren Elementen wechselt der Cursor zu einem “Verbotscursor”, wenn er sich über einem Tabellencontainer befindet (seine Fig. 4-91). Barcodefelder können zwar im Container platziert werden, werden aber nur in der ersten Zeile angezeigt.

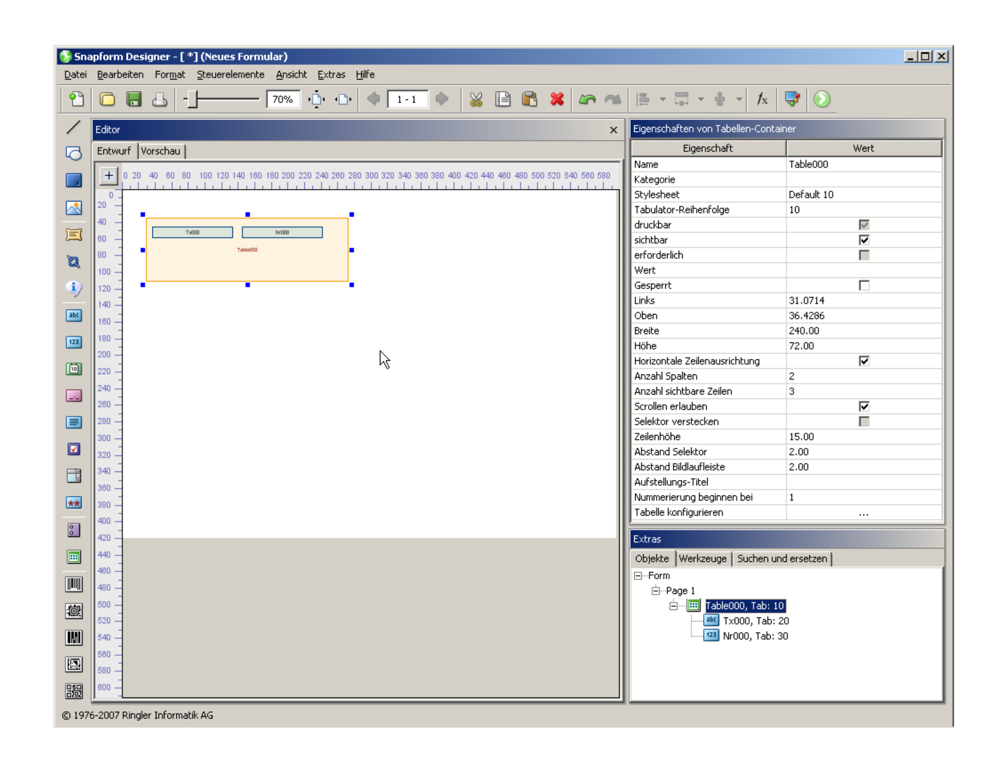
Vorsicht: *Datamatrix Barcodefelder können zwar platziert werden, können aber zu einer schwerwiegenden Korruption des Formulars führen.*

Fig. 4-138 *Verbotscursor bei nicht in einem Tabellencontainer platzierbaren Formularelementen*



Dieser Schritt wird nun so lange wiederholt, bis die Musterzeile aufgebaut ist. (siehe Fig. 4-139).

Fig. 4-139 *Tabellencontainer mit platzierten Formularelementen 092*



Die Dokumentstruktur zeigt in Beispiel die Zugehörigkeit der beiden Formularelemente **`Tx000`** und **`Dt000`** zum Tabellencontainer **`Table000`**.

Formularelemente, welche vor dem Platzieren des Tabellencontainers bereits auf dem Formular vorhanden waren, können nicht zur Gruppe der Tabellenelemente hinzugefügt werden. Ebensowenig können später ausserhalb des Bereichs des Tabellencontainers platzierte Feldelemente hinzugefügt werden.

Zu einem Tabellencontainer gehörende Feldelemente müssen innerhalb dessen Perimeter platziert sein, damit sie ordnungsgemäss angezeigt werden. Sie können auch nicht aus dem Tabellencontainer heraus verschoben werden.

Wenn ein Tabellencontainer verschoben oder skaliert wird, behalten die zugehörigen Elemente ihre relative Position bei. Das bedeutet, dass es ausreicht, den Tabellencontainer zu verschieben.

Praxistip: *Da jedes Element, welches — versehentlich oder absichtlich — innerhalb eines Tabellencontainers platziert wird, zur Tabelle hinzugefügt wird, könnte es sich empfehlen, während der weiteren Arbeit mit dem Formular den Tabellencontainer zu sperren (Eigenschaft* **Gesperrt** *gewählt).*

Nach dem Erstellen der Musterzeile (welche aus mehreren visuellen Zeilen bestehen kann), muss nun die Tabelle konfiguriert werden.

Die nachfolgenden Schritte sind für mehrzeilige Tabellen gedacht (aus einer einzigen Zeile aufgebaute "Phantom-Tabellen" werden weiter unten behandelt):

Praxistip: *Es empfiehlt sich, die Resultate der in den folgenden Schritten beschriebenen Aktionen regelmässig in der Vorschau zu kontrollieren.*

1. Den Wert für **Zeilenhöhe** auf die effektiv benötigte Zeilenhöhe abstimmen.

   Der Wert von Zeilenhöhe muss grösser sein, als die Höhe der Elemente der logischen Zeile, damit bei der Darstellung keine Überschneidungen von Elementen auftreten.

   Dies erfolgt am einfachsten durch Selektion aller Elemente innerhalb des Tabellencontainers, Kopieren und wieder Einfügen, und die immer noch aktive Selektion mit **Bearbeiten —> Gruppieren** zu einer Gruppe zusammenfügen. Das Eigenschaften-Fenster gibt nun Auskunft über die Dimensionen der Gruppe.

Praxistip: *Es ist möglich, ein zweites Snapform Designer Fenster mit einem leeren Formular offen zu halten, und die kopierten Elemente in diesem Fenster einzufügen. Damit bleibt das eigentliche Formular frei von eigentlich überflüssigen Elementen.*

2. **Höhe** des Tabellencontainers, **Zeilenhöhe** und **Anzahl sichtbare Zeilen** aufeinander abstimmen.

   Die Höhe des Tabellencontainers muss grösser sein, als Anzahl sichtbare Zeilen mal Zeilenhöhe, damit die letzte Zeile nicht angeschnitten dargestellt wird. Andererseits sollte sie nur unwesentlich grösser sein, da sonst leerer Raum entsteht, und ein allfälliger Rollbalken unübersichtlich dargestellt wird.

3. Selektor einstellen

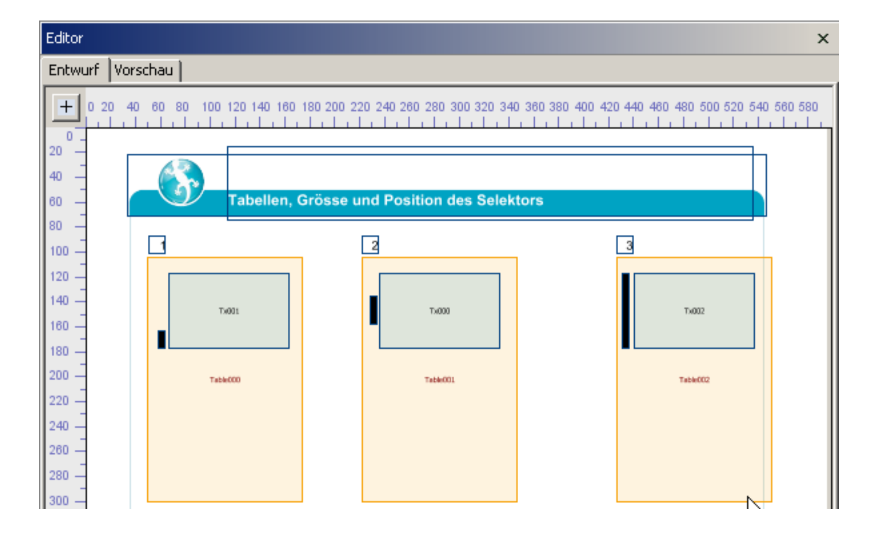
   Der Selektor ist ein nur in der Vorschau bzw. im Snapform Viewer angezeigtes visuelles Tabellenelement. Es dient einerseits zur Anzeige der Zeilennummer (für wachsende Tabellen wichtig), andererseits auch zur Selektion der aktiven Zeile. Die Grösse des Selektors entspricht der Höhe des Tabellenelements, welches die erste (sich am weitesten links befindende) Spalte ausmacht.

Die erste einstellbare Grösse für den Selektor ist der Abstand vom Tabellencontainer (**Abstand Selektor**). Der Grundwert von 2 Pt ist normalerweise sinnvoll.

Die zweite einstellbare Grösse für den Selektor ist die Nummer der ersten Zeile (**Nummerierung beginnen bei)**. Normalerweise ist dieser Wert 1, so dass die Zeilen mit 1 beginnend in aufsteigender Folge numeriert werden. Der Wert kann jedoch beliebig gesetzt werden.

Die Grösse und relative Position in der Zeile des Selektors kann beeinflusst werden, indem als erste Spalte ein sehr schmales Beschriftungsfeld mit der für den Selektor gewünschten Höhe und Position eingefügt wird, wie dies in dem Beispieldokument table_selectors.qdf dargestellt ist (siehe Fig. 4-140 (Entwurfsansicht bzw. Fig. 4-141 (Vorschau-Ansicht)).

Fig. 4-140 *Grösse und Position des Selektors in der Entwurfsansicht 093*



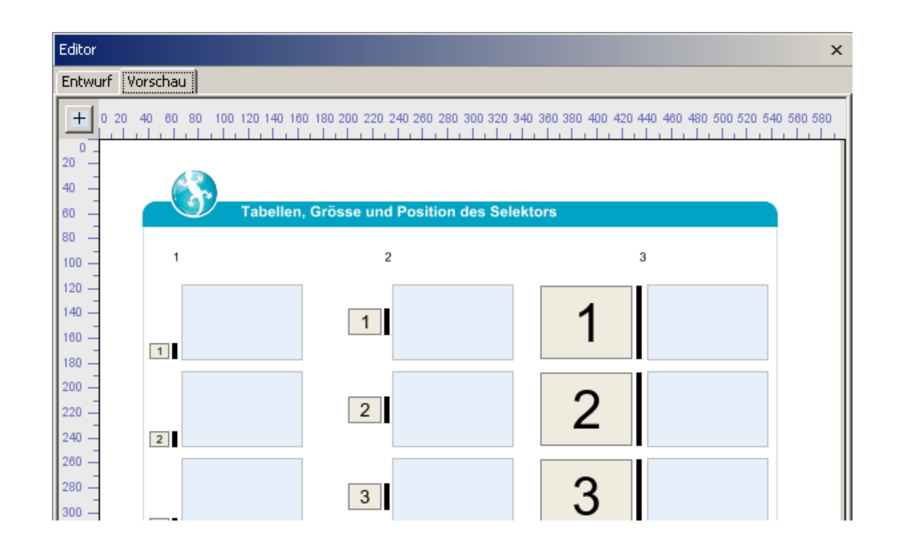


Fig. 4-141 *Grösse und Position des Selektors in der Vorschau-Ansicht 094*

1 Tabelle Table 000: Beschriftungsfeld Lb000, Höhe 13 Pt, Position bündig mit unterem Rand von Textfeld Tx001 (Höhe 60 Pt)
2 Tabelle Table001: Beschriftungsfeld Lb001, Höhe 21.7 Pt, Position mittenzentriert mit Textfeld Tx000 (Höhe 60 Pt)
3 Tabelle Table 002: Beschriftungsfeld Lb002, Höhe 60 Pt, Position bündig mit oberem rand von textfeld Tx002 (Höhe 60 Pt)

Dieses Beispiel zeigt drei sonst identisch definierte Tabellen, mit unterschiedlicher Höhe und Position des ersten Zeilenelements (einem schmalen Beschriftungsfeld, welches zur Anzeige des Sachverhalts schwarz eingefärbt worden ist). Dies erlaubt eine weitgehende Kontrolle des Selektors.

Die nachfolgenden Schritte sind nur notwendig, wenn die Eigenschaft **Scrollen erlauben** aktiv ist.

4. Position des Rollbalkens einstellen

Der Rollbalken erscheint ausserhalb des Tabellencontainers an dessen rechten Rand. Der in der Eigenschaft **Abstand Bildlaufleiste** definierte Abstand von 2 Pt vom Rand ist ein guter Erfahrungswert. Gegebenenfalls muss die Höhe und Breite des Tabellencontainers angepasst werden, damit der Rollbalken nicht allzu losgelöst von der Tabelle erscheint.

Das Aussehen des Rollbalkens wird durch das Betriebssystem bestimmt, und kann durch Snapform nicht weiter beeinflusst werden.

5. Detail-Aufstellung spezifizieren

Wenn die Tabelle mehr Zeilen enthält, als zur Anzeige vorgesehen sind, wird beim Ausdruck ein Zusatzblatt erzeugt. Dieses Zusatzblatt wird “Aufstellung” genannt. Diese Aufstellung ist eine sehr einfach gestaltete Tabelle, welche konfiguriert werden muss.

Die Aufstellung wird nur in der Druckvorschau angezeigt. Da Snapform Designer die Druckvorschau nicht erzeugen kann, muss das Formular gespeichert, und anschliessend im Snapform Viewer angezeigt werden. Dort kann mit Anklicken des Druckvorschau-Werkzeugs die Druckvorschau erzeugt werden, und mit Blättern auf Folgeseiten erscheint die Aufstellung.

Die erste Eigenschaft der Aufstellung ist der Titel, welcher in **Aufstellungs-Titel** eingegeben wird.

Der weitere Teil der Eingaben erfolgt im Eingabefenster **Tabelle konfigurieren** (siehe Fig. 4-136).

In der Aufstellungstabelle sollten die Eingaben für die Spaltenbreite kontrolliert werden. In einer unveränderten Tabelle werden die Breiten der jeweiligen Spalten-Felder übernommen. Dies hat zur Folge, dass die Spalten der Aufstellung proportional gleich breit sind wie die Spalten im Formular. Es kann aber unter Umständen sinnvoll sein, in der Aufstellung Spaltenbreiten zu verändern (zum Beispiel die Breite für das zur Kontrolle des Selektors eingefügte Beschreibungsfeld).

In jedem Fall sollten Spaltentitel eingegeben werden. Während diese Titel im Formular mit Sicherheit vorhanden sind, müssen sie in der Aufstellung explizite in der Spalte **Kopfzeile** eingegeben werden.

Das Kontrollkästchen **Summe anzeigen** kann angeklickt werden. In diesem Fall wird, sofern es der Feldtyp der Spalte zulässt, die Summe über die Zeilen gebildet, und unterhalb der Aufstellung angezeigt. Ob dies erwünscht ist, hängt von der jeweiligen Anwendung ab.

Die **Spalte für Nachricht** sollte wenn immer möglich festgelegt werden, da sonst im Formular kein Hinweis auf die Seite der Aufstellung erfolgt. Hier ist zu beachten, dass es für die Druckvorschau keine Rolle spielt, ob das Feld der Spalte den Text "siehe Aufstellung" anzeigen kann (Datums- oder Zahlenfelder funktionieren also). Das Feld muss aber breit genug sein, dass der ganze Text angezeigt werden kann. Die für den Text benutzte Schriftart wird durch das Stylesheet des Spaltenfelds gesteuert.

Für die Tabellenüberschrift und die Spaltentitel ist ein passendes Stylesheet zu wählen (**Stylesheet für Überschrift**, bzw. **Stylesheet für Kopfzeile**). Grundwert hier ist das Default-Stylesheet.

Die Aufstellung wird, wenn nicht weiter spezifiziert, nach der letzten Seite des Formulars gedruckt. Wenn sie an einer anderen Stelle erscheinen soll, wird die Seite, nach welcher die Aufstellung erscheinen soll, mit **Druck nach Seite** gewählt.

Schliesslich ist durch Anklicken des betreffenden Optionsfelds von **Seitenausrichtung** noch anzugeben, ob die Aufstellung im Hoch- oder Querformat ausgedruckt werden soll.

Die Zeilenhöhe in der Aufstellung richtet sich nach dem Wert der **Zeilenhöhe** in der Tabelle, welche proportional zur Tabellenbreite skaliert wird. Damit ist sicher gestellt, dass die Aufstellung zumindest gleich viel Information darstellen kann, wie die Tabelle im Formular.

Wenn die Tabelle eine "Phantom-Tabelle" ist, wird im Formular selbst nur eine einzige Zeile angezeigt. Dies hat zur Folge, dass auch die

Konfiguration der Tabelle besonders betrachtet werden muss. Eine "Phantom-Tabelle wird folgendermassen konfiguriert:

Praxistip: *Es empfiehlt sich, die Resultate der in den folgenden Schritten beschriebenen Aktionen regelmässig in der Vorschau zu kontrollieren.*

1. Den Wert für **Zeilenhöhe** auf die effektiv benötigte Zeilenhöhe abstimmen.

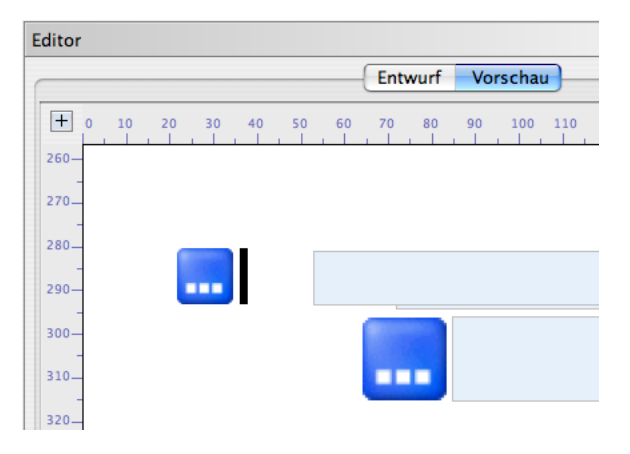
   Der Wert von Zeilenhöhe muss grösser sein, als die Höhe der Elemente der logischen Zeile, damit bei der Darstellung keine Überschneidungen von Elementen auftreten.

   Da nur eine einzige Zeile angezeigt wird, wird sich der Wert der Zeilenhöhe in der Gegend der Höhe des Tabellencontainers bewegen.

2. Schaltflächen zum Öffnen der Dateneingabetabelle aktivieren

   Da die Tabelle nur eine einzige Zeile anzeigt, ist die Dateneingabe eingeschränkt. Die eigentliche Dateneingabe und auch Anzeige am Bildschirm erfolgt über die Dateneingabetabelle. Diese Tabelle wird durch Anklicken einer Schaltfläche geöffnet (siehe Fig. 4-142).

Fig. 4-142 *Schaltfläche zum Öffnen der Dateneingabetabelle (stark vergrössert)*



   Diese Schaltfläche wird aktiviert, indem in der Aufstellungs-Tabelle im Tabelle konfigurieren Dialog eine oder mehrere Kontrollkästchen **Summe anzeigen** eingeschaltet werden.

Die Schaltfläche erscheint links des selektierten Feldelements. Die Grösse richtet sich nach der Höhe des betreffenden Feldelements, wie das Beispiel in Fig. 4-142 zeigt.

3. Dateneingabetabelle spezifizieren

Die Dateneingabetabelle ist eine rein Bildschirm-bezogene Tabelle, und kann nicht ausgedruckt werden (ihr Äquivalent ist die Aufstellung). Sie kann in einem gewissen Rahmen spezifiziert werden. Verschiedene Eigenschaften werden von der Dateneingabetabelle und der Aufstellung gemeinsam genutzt.

Die erste Eigenschaft der Dateneingabetabelle ist der Titel, welcher in **Aufstellungs-Titel** eingegeben wird.

Der weitere Teil der Eingaben erfolgt im Eingabefenster **Tabelle konfigurieren** (siehe Fig. 4-136).

In der Aufstellungstabelle werden die Spaltenbreiten der Eingabespalten gesteuert. In einer unveränderten Tabelle werden die Breiten der jeweiligen Spalten-Felder übernommen. Dies hat zur Folge, dass die Spalten der Aufstellung proportional gleich breit sind wie die Spalten im Formular. Es kann aber unter Umständen sinnvoll sein, in der Aufstellung Spaltenbreiten zu verändern (zum Beispiel die Breite für allfällig eingefügte Beschreibungsfelder).

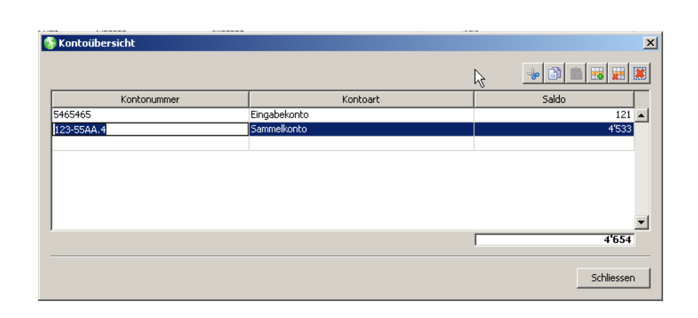
Bei Tabellen mit mehrzeiligen logischen Zeilen ist zu beachten, dass die Numerierung der Spalten strikt von links nach rechts erfolgt, und keine Rücksicht auf die visuelle Zeile des betreffenden Felds nimmt. Es ist daher umso wichtiger, Spaltentitel eindeutig zu definieren (siehe unten).

In jedem Fall sollten Spaltentitel eingegeben werden. Während diese Titel im Formular mit Sicherheit vorhanden sind, müssen sie für die Dateneingabetabelle explizite in der Spalte **Kopfzeile** eingegeben werden.

Das Kontrollkästchen **Summe anzeigen** kann angeklickt werden. Es bedeutet einerseits, dass in der Bildschirmanzeige die Schaltfläche zum Öffnen der Dateineingabetabelle erscheint, andererseits in der Dateneingabetabelle die laufende Spaltensumme für die betreffende Tabelle angezeigt wird, welche ebenfalls in der Aufstellung erscheint. Die Summe wird nur dann angezeigt, wenn es der Feldtyp der Spalte zulässt.

Damit ist die Dateneingabetabelle spezifiziert. In der Vorschau-Ansicht erscheint nun die Schaltfläche zum öffnen. Das in Fig. 4-143 gezeigte Beispiel stammt von der Musterdatei tabellen.qdf.

Fig. 4-143 *Dateneingabetabelle für Position 3 von tabellen.qdf 095*



Die Eingaben in die Dateneingabetabelle erfolgen wie in einer üblichen Tabellenkalkulation.

Die Dateneingabetabelle hat eine aus 6 Werkzeugen bestehende Werkzeugleiste (siehe Fig. 4-144).

Fig. 4-144 *Werkzeugleiste der Dateneingabetabelle*



Ausschneiden

Das Ausschneidewerkzeug entfernt die aktuelle Zeile und legt sie in der Zwischenablage ab. Die nachfolgenden Zeilen rücken nach oben nach.

Kopieren

Das Kopierwerkzeug kopiert die aktuelle Zeile in die Zwischenablage.

Einfügen

Das Einfügewerkzeug fügt die in der Zwischenablage gespeicherte Zeile am Ende der Tabelle ein.

Zeile hinzufügen

Das Zeile hinzufügen-Werkzeug fügt oberhalb der aktiven Zeile eine neue leere Zeile hinzu.

Zeile entfernen

Das Zeile hinzufügen-Werkzeug fügt oberhalb der aktiven Zeile eine neue leere Zeile hinzu.

Löschen

Das Tabelle löschen-Werkzeug löscht den Inhalt der Dateneingabetabelle und entfernt sämtliche Zeilen.

4. Detail-Aufstellung spezifizieren

Bei "Phantom-Tabellen" wird beim Ausdruck ein Zusatzblatt erzeugt. Dieses Zusatzblatt wird "Aufstellung" genannt. Diese Aufstellung ist eine sehr einfach gestaltete Tabelle, welche konfiguriert werden muss. Gewisse Eigenschaften werden mit der Dateineingabetabelle gemeinsam benutzt, und brauchen nicht mehr neu konfiguriert zu werden.

Die Aufstellung wird nur in der Druckvorschau angezeigt. Da Snapform Designer die Druckvorschau nicht erzeugen kann, muss das Formular gespeichert, und anschliessend im Snapform Viewer angezeigt werden. Dort kann mit Anklicken des Druckvorschau-Werkzeugs die Druckvorschau erzeugt werden, und mit Blättern auf Folgeseiten erscheint die Aufstellung.

Die Titel der Aufstellung ist bereits für die Dateineingabetabelle in **Aufstellungs-Titel** spezifiziert worden.

Der weitere Teil der Eingaben erfolgt im Eingabefenster Tabelle konfigurieren (siehe Fig. 4-136).

Die Spaltenbreiten für die Aufstellung sind identisch mit denjenigen für die Dateineingabetabelle, und sind dort bereits definiert worden. Ebenso die Spaltentitel in der Spalte **Kopfzeile**

und die Wahl der Spalten mit Summen durch das Kontrollkästchen **Summe anzeigen**.

Die **Spalte für Nachricht** sollte wenn immer möglich festgelegt werden, da sonst im Formular kein Hinweis auf die Seite der Aufstellung erfolgt. Hier ist zu beachten, dass es für die Druckvorschau keine Rolle spielt, ob das Feld der Spalte den Text "siehe Aufstellung" anzeigen kann (Datums- oder Zahlenfelder funktionieren also). Das Feld muss aber breit genug sein, dass der ganze Text angezeigt werden kann. Die für den Text benutzte Schriftart wird durch das Stylesheet des Spaltenfelds gesteuert.

Für die Tabellenüberschrift und die Spaltentitel ist ein passendes Stylesheet zu wählen (**Stylesheet für Überschrift**, bzw. **Stylesheet für Kopfzeile**). Grundwert hier ist das Default-Stylesheet.

Die Aufstellung wird, wenn nicht weiter spezifiziert, nach der letzten Seite des Formulars gedruckt. Wenn sie an einer anderen Stelle erscheinen soll, wird die Seite, nach welcher die Aufstellung erscheinen soll, mit **Druck nach Seite** gewählt.

Schliesslich ist durch Anklicken des betreffenden Optionsfelds von **Seitenausrichtung** noch anzugeben, ob die Aufstellung im Hoch- oder Querformat ausgedruckt werden soll.

Die Zeilenhöhe in der Aufstellung richtet sich nach dem Wert der **Zeilenhöhe** in der Tabelle, welche proportional zur Tabellenbreite skaliert wird. Damit ist sicher gestellt, dass die Aufstellung zumindest gleich viel Information darstellen kann, wie die Tabelle im Formular.

Damit ist die Tabelle definiert, und das Formular kann eingesetzt werden.

4.4.7.3 Tabellen-Beispiele

Die nachstehenden Beispiele zeigen die verschiedenen Erscheinungsformen von Tabellen in Snapform.

Im Dokument tabellen.qdf (siehe Fig. 4-145 (Entwurfsmodus)) bzw. Fig. 4-146ff (Ansicht in Snapform Viewer)) sind verschiedene Beispiele von Tabellen gezeigt.

Nach dem Selektieren der Feldelemente im Snapform Designer gibt das Eigenschaften-Fenster Auskunft über dessen jeweiligen Merkmale.

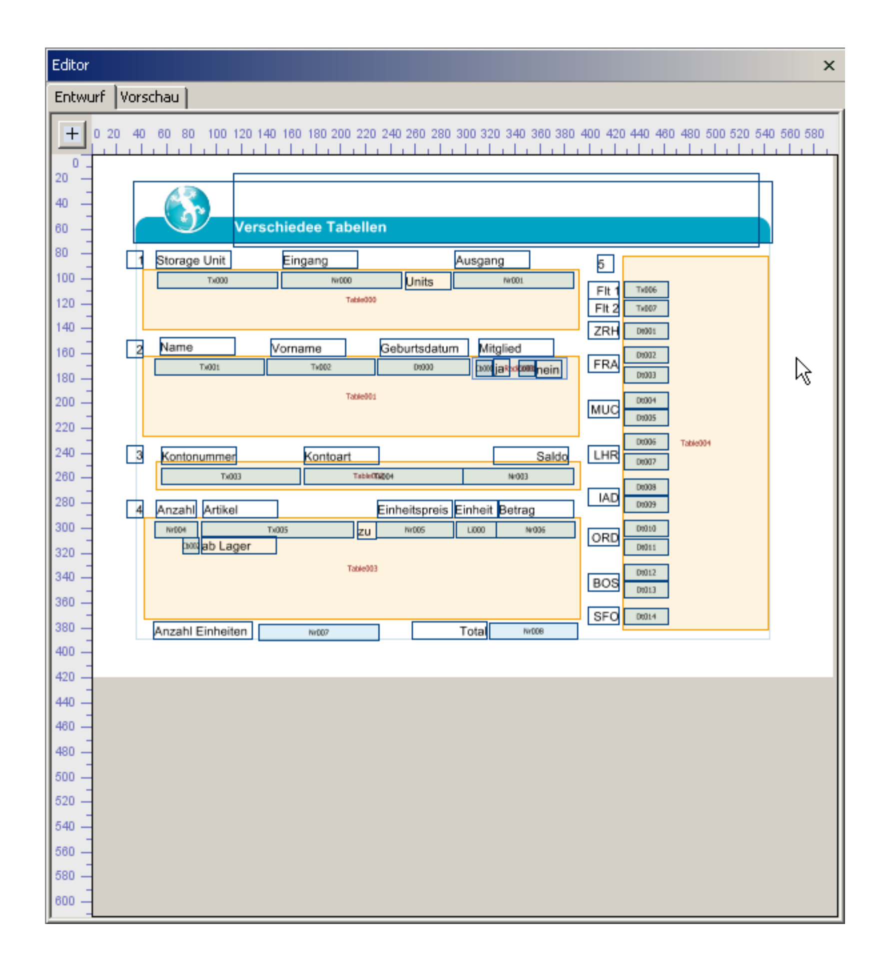
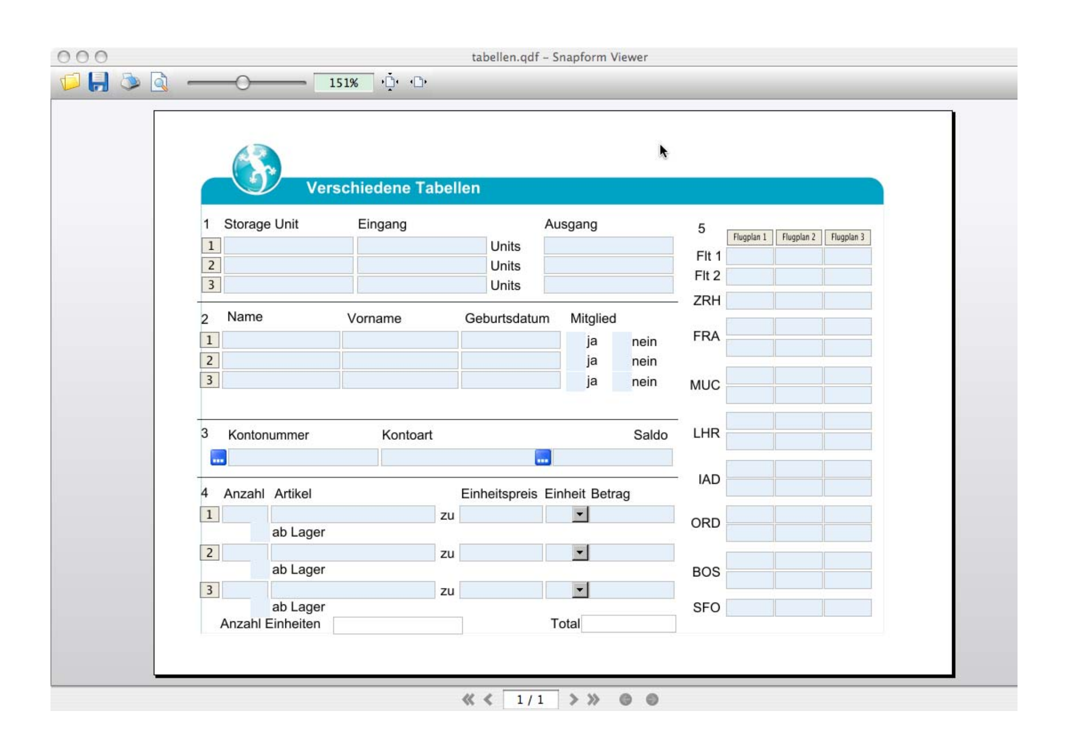


Fig. 4-145 *Beispiele von tabellen.qdf, im Entwurfs-Modus 096*

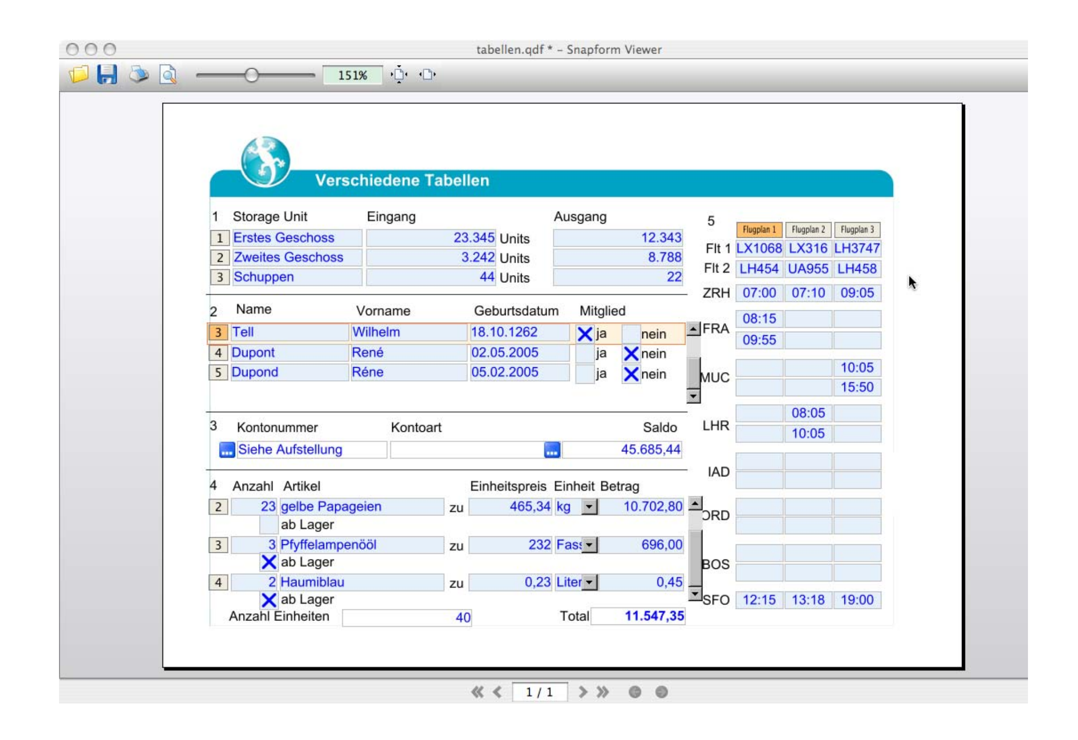
Fig. 4-146 *Beispiele von tabellen.qdf in Snapform Viewer, unausgefüllt*



1 Normale Tabelle, fixe Anzahl Zeilen
2 Tabelle mit variabler Anzahl Zeilen, und 3 angezeigten Zeilen
3 Phantom-Tabelle mit einer angezeigten Zeile
4 Tabelle mit variabler Anzahl Zeilen, mehrere visuelle Zeilen pro logische Zeile, Summenbildung unterhalb.
5 Tabelle mit variabler Anzahl Zeilen, vertikale logische Zeilen

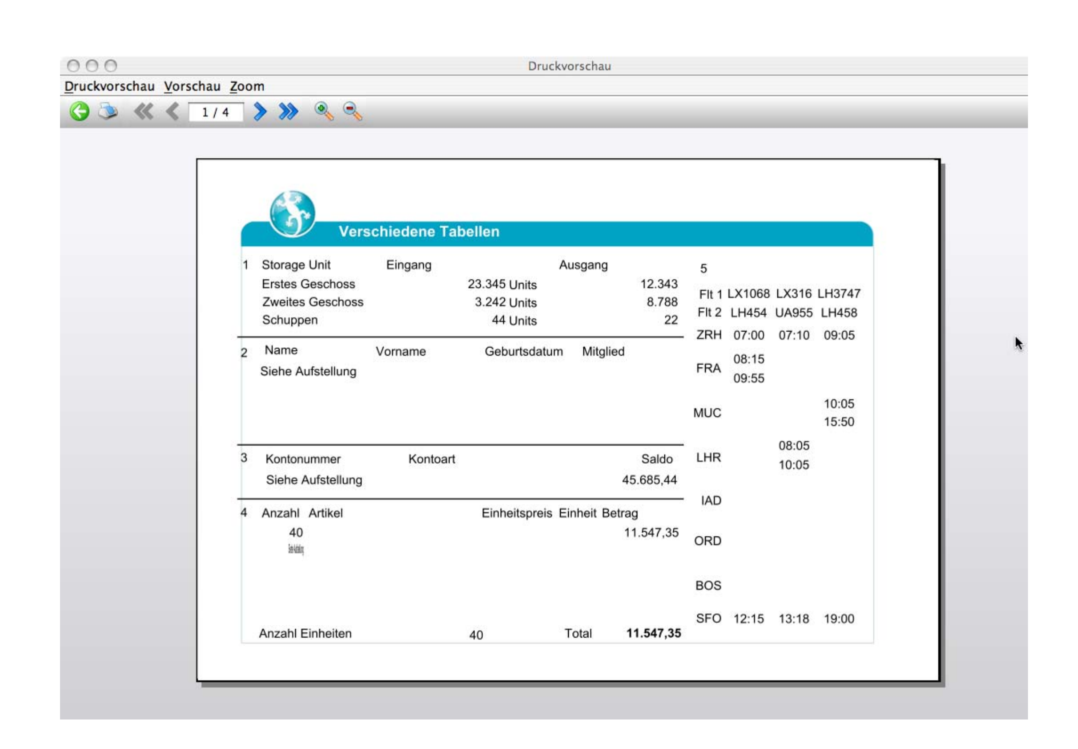
In Fig. 4-147 ist dasselbe Formular nochmals gezeigt, aber mit ausgefüllten Tabellenfeldern.

Fig. 4-147 *Beispiele von tabellen.qdf in Snapform Viewer, ausgefüllt 098*

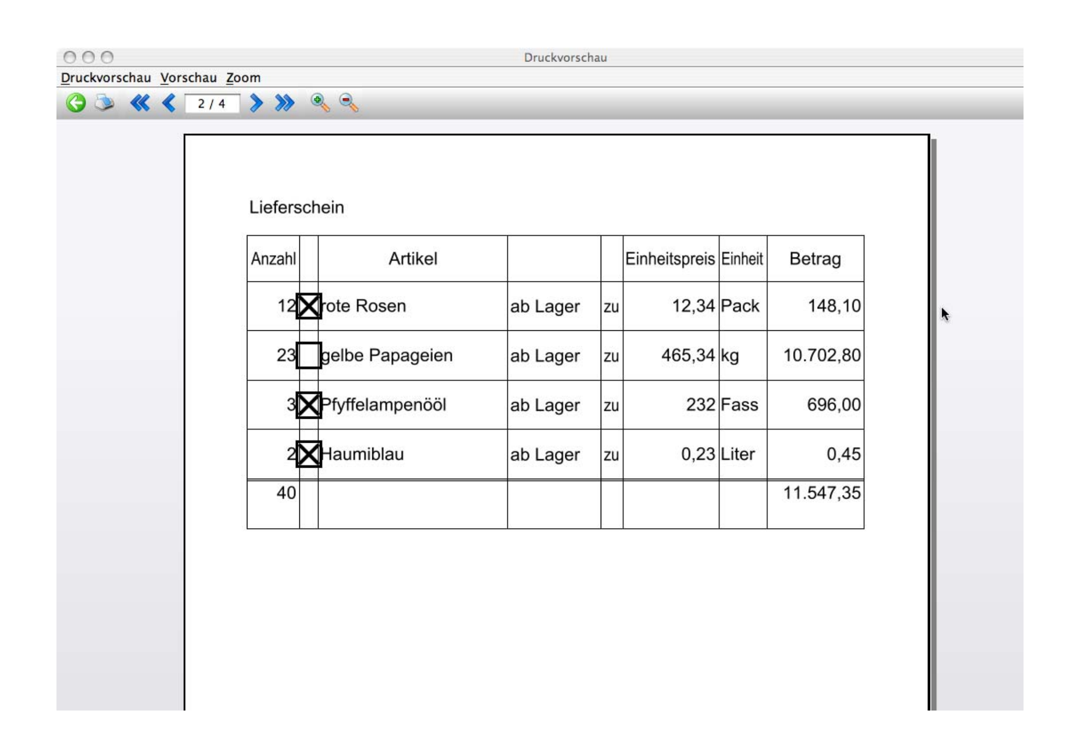


In Fig. 4-148 ist dasselbe Formular in der Druck-Voranschau gezeigt.

Fig. 4-148 *Beispiele von tabellen.qdf in Snapform Viewer, Druckvorschau 099*



In Fig. 4-148 ist schliesslich die Aufstellung von Beispiel 3 abgebildet.



Lieferschein

Anzahl		Artikel			Einheitspreis	Einheit	Betrag
12	☒	rote Rosen	ab Lager	zu	12,34	Pack	148,10
23	☐	gelbe Papageien	ab Lager	zu	465,34	kg	10.702,80
3	☒	Pfyffelampenööl	ab Lager	zu	232	Fass	696,00
2	☒	Haumiblau	ab Lager	zu	0,23	Liter	0,45
40							11.547,35

Fig. 4-149 *Aufstellung von Beispiel 4 von tabellen.qdf in Snapform Viewer 100*

4.4.8 Aktive Elemente

Aktive Elemente sind Formularelemente, welche primär als Mittel für Aktionen des Benutzers dienen, sonst jedoch für das Formular an sich keine weitere Bedeutung haben. Typische Beispiele für aktive Elemente sind Schaltflächen oder Links.

Snapform kennt zwei Typen von aktiven Elementen, welche in einem Formular platziert werden können: Hyperlinks und Hinweise. Zusätzlich kann das Formular über die Formulareinstellungen im Viewer eine Reihe von Befehlstasten aktivieren.

4.4.8.1 Hyperlinks

Hyperlinks sind Formularelemente, welche den Benutzer an einen anderen Ort bringen. Dies kann einerseits ein anderes Formularelement sein, andererseits kann eine Internet-Adresse aufgerufen werden, welche dann durch die entsprechende Software (Webbrowser, Mail-Client) angezeigt wird.

Hyperlinks können zum Aufbau eines erweiterten Referenz- und Hilfesystems benutzt werden, bei welchem im Intra- oder Internet zugängliche Informationen aufgerufen werden.

Das Hyperlinkwerkzeug erzeugt eine aktive Fläche, welche als Absprungpunkt für den Hyperlink dient. Nach Wahl des Werkzeugs erscheint ein Cursor mit einem Hypyerlink in Default-Grösse (siehe Fig. 4-150).

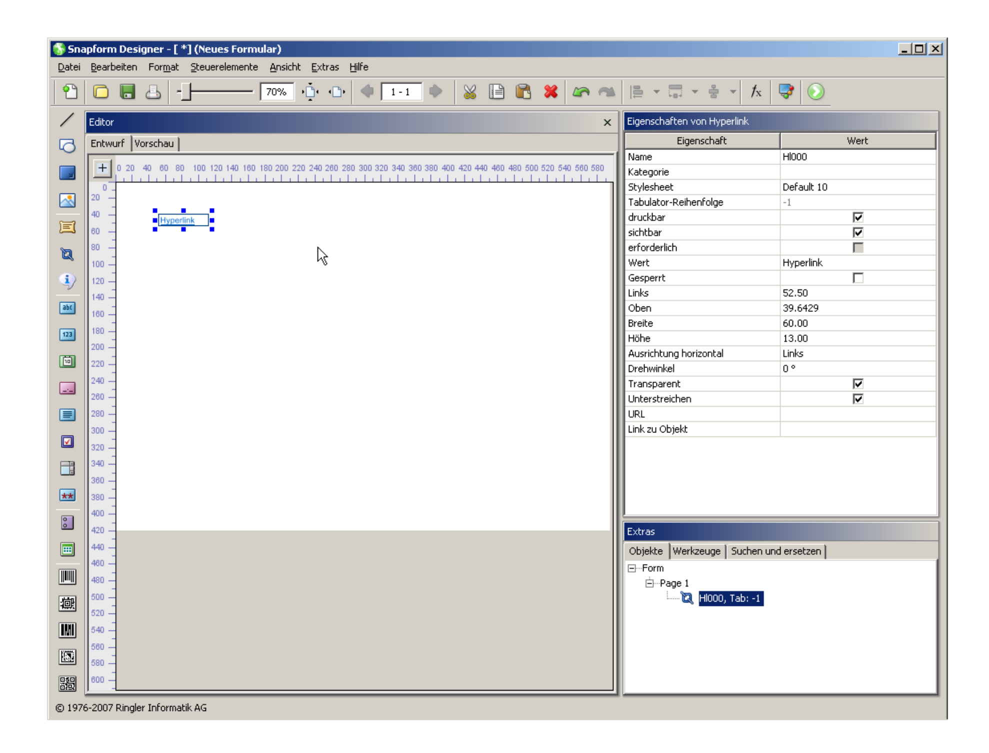
Fig. 4-150 *Hyperlink-Cursor*



Mit dem Hyperlink-Werkzeugt verknüpft ist ein Hyperlink-Feld, 60 pt breit und 13 pt hoch. Mit Mausklick wird dieses Feld auf der Arbeitsfläche platziert. Dabei erhält es automatisch einen Namen, bestehend aus den Zeichen Hl und einer dreistelligen fortlaufenden Nummer, beginnend mit 000. Das erste in einem Formular mit dem Kontrollkästchen-Werkzeug platzierte Feld trägt daher den Namen Hl000. In der Objektstruktur wird es der betreffenden Seite hinzugefügt. In Fig. 4-151 ist der erste Hyperlink in einem neuen Formular platziert, und anschliessend mit Anklicken selektiert worden.

Zur besseren Kennzeichnung und als Erinnerung für das Anpassen trägt ein neu platzierter Hyperlink den Text **Hyperlink**, welcher die Feldbezeichnung überdeckt.

Fig. 4-151 *Neu erstellter Hyperlink 101*



Die Länge und Breite des Hyperlinks kann durch Ziehen der Ankerpunkte des Selektionsrechtecks verändert werden, die Position auch durch Anfassen und Ziehen. Beim Überfahren der Ankerpunkte ändert der Cursor in Doppelpfeile, welche die Bewegungsrichtung des Ankerpunktes angeben. Beim “Anfassen” und Drücken der Maustaste ändert sich der Cursor zu einem Hohlpfeil.

Die übrigen Merkmale eines Hyperlinks werden über das Eigenschaften-Fenster kontrolliert (siehe Fig. 4-152).

Fig. 4-152 *Eigenschaften eines Hyperlinks 101*

Eigenschaften von Hyperlink

Eigenschaft	Wert
Name	Hl000
Kategorie	
Stylesheet	Default 10
Tabulator-Reihenfolge	-1
druckbar	☑
sichtbar	☑
erforderlich	☐
Wert	Hyperlink
Gesperrt	☐
Links	52.50
Oben	39.6429
Breite	60.00
Höhe	13.00
Ausrichtung horizontal	Links
Drehwinkel	0 °
Transparent	☑
Unterstreichen	☑
URL	
Link zu Objekt	

Name

Der Name des Hyperlinks im Dokument. Dieser Name kann geändert werden.

Kategorie

Zugehörigkeit zu einer oder mehreren Kategorien, welche es erlauben, logisch zusammengehörende Gruppen von Elementen zu bilden. Beim Anklicken dieses Felds öffnet sich der Kategorien Dialog (siehe Fig. 4-23).

Die Kategorien werden durch Anklicken des betreffenden Kontrollkästchens ausgewählt. Mit **OK** wird die Auswahl übernommen.

Stylesheet

Das für das Element gültige Stylesheet. Wenn explizite kein Stylesheet ausgewählt worden ist, wird das erste Default-Stylesheet genutzt.

Tabulator-Reihenfolge

Ordnungsnummer des Elements in der Tabulator-Reihenfolge. Dieser Wert ist immer -1 und kann nicht verändert werden. Das bedeutet, dass ein Hyperlink nicht Bestandteil der Tabulatoren-Reihenfolge ist.

druckbar

Wenn dieses Kontrollkästchen gewählt ist, wird der Hyperlink (dessen Text) ausgedruckt.

sichtbar

Wenn dieses Kontrollkästchen gewählt ist, wird der Hyperlink auf dem Bildschirm angezeigt.

erforderlich

Diese Eigenschaft ist bei Hyperlinks immer ausgeschaltet, und kann nicht geändert werden.

Wert

Dies ist der Text, welcher im Hyperlink angezeigt wird. In der Grundeinstellung lautet dieser Text Hyperlink.

Wenn diese Eigenschaft angeklickt wird, öffnet sich der Formeleditor zur Texteingabe. Dies kann entweder ein konstanter Text sein (wie die Grundeinstellung), oder aber das Resultat einer Formel (was den Aufbau dynamischer URL erlaubt).

Es werden nur einfache Texte unterstützt; HTML-Code wird wörtlich wiedergegeben.

Gesperrt

Wenn dieses Kontrollkästchen gewählt ist, wird der Hyperlink im Snapform Designer blockiert, und kann nicht (versehentlich) verschoben werden. Zudem werden sämtliche übrigen Eigenschaften im Eigenschaften-Dialog deaktiviert. Der Zugriff auf die Eigenschaften ist erst wieder möglich, wenn dieses Kontrollkästchen deaktiviert wird.

Links

X-Koordinate der linken oberen Ecke der Umrandungsbox des Elements in Points. Diese Eigenschaft erlaubt eine genaue numerische Positionierung des Hyperlinks.

Oben

Y-Koordinate der linken oberen Ecke der Umrandungsbox des Elements in Points. Diese Eigenschaft erlaubt eine genaue numerische Positionierung des Hyperlinks.

Breite

Breite der Umrandungsbox des Hyperlinks.

Höhe

Höhe der Umrandungsbox des Hyperlinks.

schreibgeschützt

Wenn dieses Kontrollkästchen gewählt ist, ist das Kontrollkästchen als schreibgeschützt markiert, und dessen Inhalt kann nicht manuell überschrieben werden. Der Wert kann nach wie vor über Formeln verändert werden. Ein schreibgeschütztes Kontrollkästchen ist nach wie vor in der Tab-Reihenfolge eingebunden, und dessen Formeln werden auch entsprechend des Zustands ausgeführt.

Ausrichtung horizontal

Diese Eigenschaft bestimmt, wie der Text im Hyperlinkfeld ausgerichtet wird.

Wenn der Eigenschaftswert angeklickt wird, erscheint ein Drop Down Menu mit der Auswahl **links**, **zentriert**, **rechts**, welche den Text linksbündig, mittenzentriert bzw. rechtsbündig im Hyperlinkfeld positioniert. Grundeinstellung ist **links**.

Drehwinkel

Der Text im Hyperlinkfeld kann um Vielfache von 90° gedreht werden. Beim Anklicken dieser Eigenschaft öffnet sich ein Drop Down Menu mit der Auswahl der Drehwinkel **0°**, **90°**, **180°**, **270°**. Grundeinstellung ist **0°**.

Transparent

Der Hintergrund des Hyperlinkfelds wird mit diesem Kontrollkästchen auf transparent (keine deckende Hintergrundsfarbe) geschaltet. Diese Eigenschaft ist in der Grundeinstellung eingeschaltet.

Unterstreichen

Hyperlinks sind im Web-Umfeld meistens unterstrichen. Der Text des Hyperlinkfelds wird mit diesem Kontrollkästchen auf unterstrichen geschaltet. Diese Eigenschaft ist in der Grundeinstellung eingeschaltet.

URL

Dies ist die Adresse (URL), zu welcher der Hyperlink führt. Snapform unterstützt diejenigen URI-Protokolle, welche im jeweiligen Rechners registriert sind. Die folgenden Protokolle werden typischerweise unterstützt:

- http:, https: Webseiten
- mailto: E-Mail

- file: Datei öffnen

Hinweis: *Es ist voll und ganz in der Verantwortung des Erstellers des Formulars, sicher zu stellen, dass die Zieladresse einwandfrei und ohne Gefahren für den Anwender ist. Ringler Informatik lehnt jegliche Haftung bei allfälligen durch die Benutzung von Hyperlinks entstehenden Schäden vollumfänglich ab.*

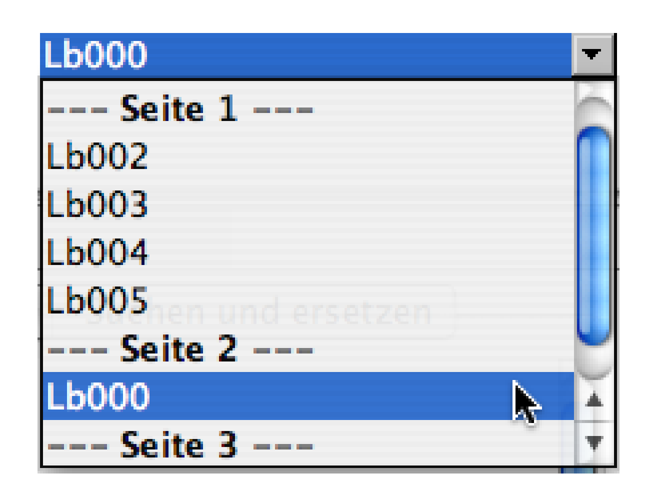
Diese Eigenschaft ist ein einfaches Eingabefeld, in welches der URL eingegeben werden kann.

Link zu Objekt

Neben Links nach externen Orten (welche über die Eigenschaft **URL** kontrolliert werden, bietet das Hyperlink-Werkzeug auch die Möglichkeit, Links innerhalb des Formulars aufzubauen. Für derartige interne Links wird diese Eigenschaft benutzt.

Wenn diese Eigenschaft gewählt wird, öffnet sich ein Drop Down-Menu mit einer Liste aller Elemente der Formularebene (alle Elemente bei welchen die Tabreihenfolge einen Wert grösser als 0 aufweist, also auch Bildfelder und Beschriftungsfelder). Die Elemente sind nach Dokumentenseite geordnet, so dass auch mehrfach vorkommende Elemente gezielt angesprungen werden können (siehe Fig. 4-153). Es kann nun ein Feld gewählt werden.

Fig. 4-153 *Auswahl von Ziel-Objekten in Link zu Objekt 102*



Im Dokument hyperlinks.qdf (siehe Fig. 4-154 (Entwurfsmodus)) bzw. Fig. 4-155ff (Ansicht in Snapform Viewer)) sind verschiedene Beispiele von Hyperlinks gezeigt.

Nach dem Selektieren der Feldelemente im Snapform Designer gibt das Eigenschaften-Fenster Auskunft über dessen jeweiligen Merkmale.

Fig. 4-154 *Beispiele von hyperlinks.qdf, im Entwurfs-Modus 103*

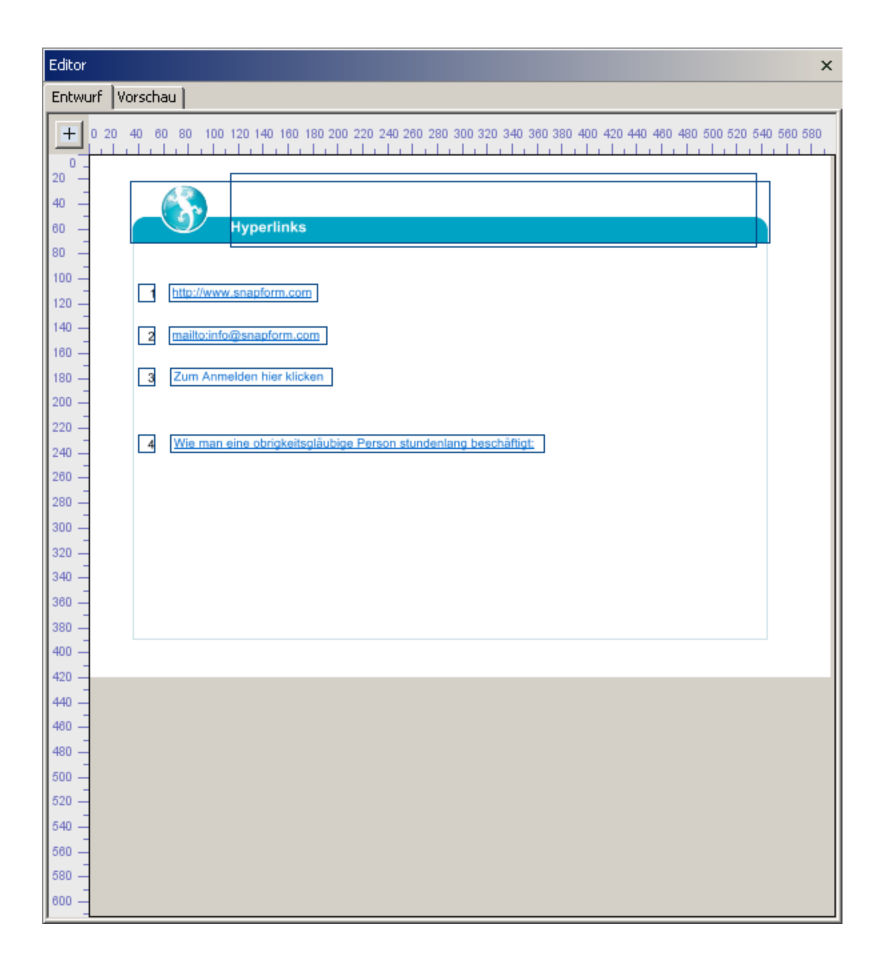
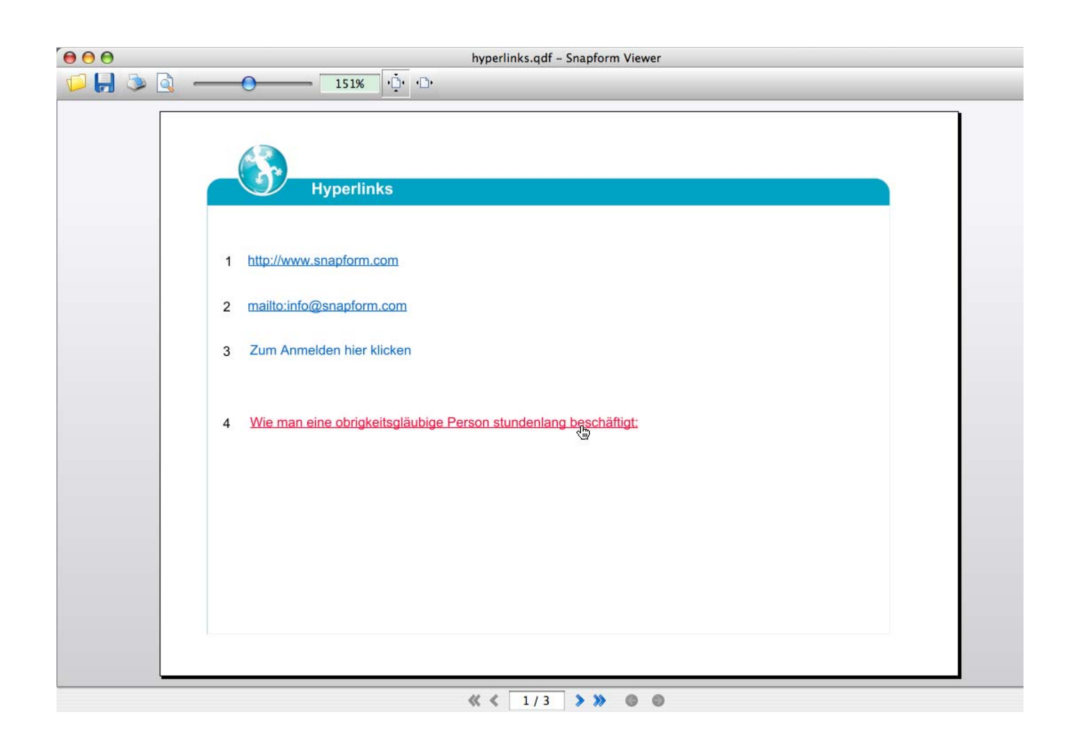


Fig. 4-155 *Beispiele von hyperlinks.qdf in Snapform Viewer*



1 Hl000: einfacher Hyperlink zu einer Website; URL: **http://www.snapform.com**
2 Hl001: Hyperlink zu Mail-Client; URL: **mailto:info@snapform.com**
3 Hl002: Hyperlink zu Mail-Client mit Vorausfüllung für Betreff und Nachricht, Unterstreichen ausgeschaltet; URL: **mailto:info@snapform.com?subject=Anmeldung&body=Ich%20melde%20mich%20an**
4 Hl003: Interner Hyperlink zu Beschreibungsfeld Lb000, welches sich auf der zweiten Seite des Formulars befindet (Auswahl in **Link zu Objekt** siehe Fig. 4-153).

4.4.8.2 Hinweise/Infopunkte

Ein grosser Vorteil von elektronischen Formularen ist, dass Zusatzinformationen und Erklärungen auf einfache Weise direkt am Ort, an dem sie benötigt werden, angezeigt werden können (ohne das Formular selbst zu überladen). Eine Möglichkeit zur Anzeige von Erklärungen besteht für Eingabefelder, in Form der Kurzinformation. Dies ist sehr sinnvoll, aber sollte nur direkt auf das Feld bezogene Informationen enthalten.

Auf der anderen Seite sind Hyperlinks, welche auf eine externe Quelle verweisen (bzw. eine entsprechend aufgebaute Seite des Dokuments).

Für den Fall, dass Zusatzinformationen nicht direkt auf ein Feld bezogen sind, und der Benutzer sich nicht von der Seite entfernen soll, kennt Snapform die sogenannten Infopunkte, oder Hinweise. Dabei handelt es sich um ein aktives Element, bei welchem beim Darüberfahren mit der Maus ein Text aufklappt, welcher entsprechende Angaben machen kann. Ein Hinweis kann daher so gestaltet werden, dass er für einen Bereich des Formulars dient. Das aktive Element selbst ist jedoch klein genug, um nicht störend zu wirken.

Das Infopunkt-Werkzeug erzeugt ein Feld, bei welchem das Darüberfahren mit der Maus einen Text anzeigt. Nach Wahl des Werkzeugs erscheint ein Cursor mit einem Infopunkt in Default-Grösse (siehe Fig. 4-156).

Fig. 4-156 *Infopunkt-Werkzeug-Cursor*



Mit dem Infopunkt-Werkzeug verknüpft ist ein Infopunkt-Feld, 12 pt breit und 12 pt hoch. Mit Mausklick wird dieses Feld auf der Arbeitsfläche platziert. Dabei erhält es automatisch einen Namen, bestehend aus den Zeichen Ip und einer dreistelligen fortlaufenden Nummer, beginnend mit 000. Das erste in einem Formular mit dem Infopunkt-Werkzeug platzierte Element trägt daher den Namen Ip000. In der Objektstruktur wird es der betreffenden Seite hinzugefügt. In Fig. 4-157 ist der erste Infopunkt in einem neuen Formulars platziert, und anschliessend mit Anklicken selektiert worden.

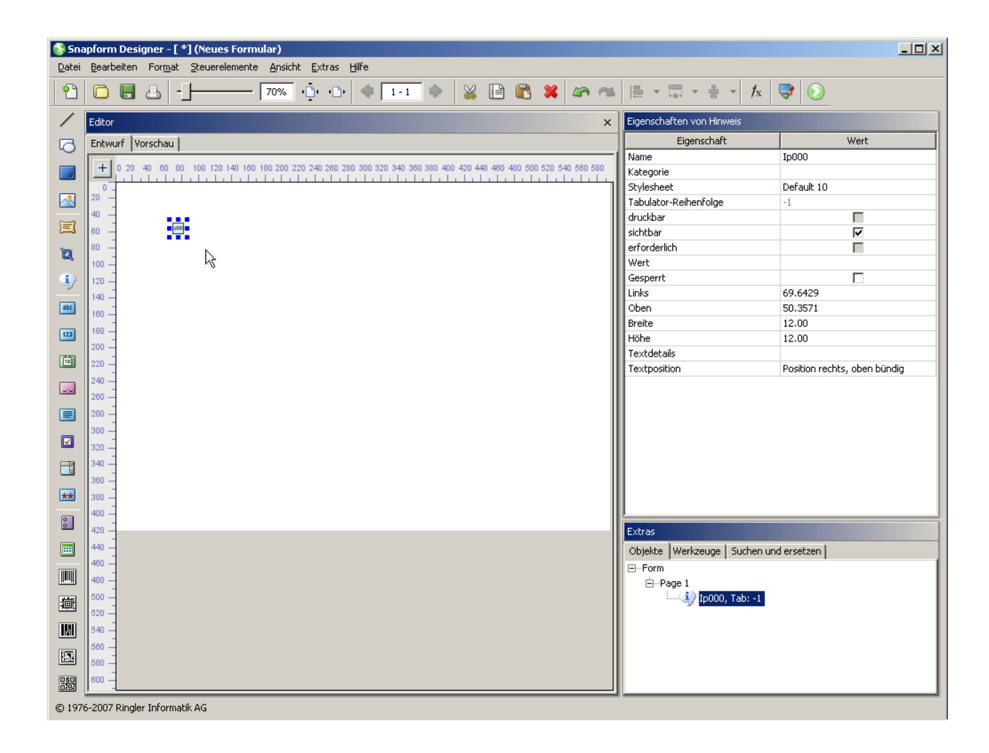


Fig. 4-157 *Neu erstellter Infopunkt 105*

Die Länge und Breite Infopunkts kann durch Ziehen der Ankerpunkte des Selektionsrechtecks verändert werden, die Position auch durch Anfassen und Ziehen. Beim Überfahren der Ankerpunkte ändert der Cursor in Doppelpfeile, welche die Bewegungsrichtung des Ankerpunktes angeben. Beim “Anfassen” und Drücken der Maustaste ändert sich der Cursor zu einem Hohlpfeil.

Die übrigen Merkmale eines Infopunkts werden über das Eigenschaften-Fenster kontrolliert (siehe Fig. 4-158).

Fig. 4-158 *Eigenschaften eines Infopunkts 105*

Eigenschaften von Hinweis

Eigenschaft	Wert
Name	Ip000
Kategorie	
Stylesheet	Default 10
Tabulator-Reihenfolge	-1
druckbar	☐
sichtbar	☑
erforderlich	☐
Wert	
Gesperrt	☐
Links	69.6429
Oben	50.3571
Breite	12.00
Höhe	12.00
Textdetails	
Textposition	Position rechts, oben bündig

Name

Der Name des Infopunkt-Felds im Dokument. Dieser Name kann geändert werden.

Kategorie

Zugehörigkeit zu einer oder mehreren Kategorien, welche es erlauben, logisch zusammengehörende Gruppen von Elementen zu bilden. Beim Anklicken dieses Felds öffnet sich der Kategorien Dialog (siehe Fig. 4-23).

Die Kategorien werden durch Anklicken des betreffenden Kontrollkästchens ausgewählt. Mit **OK** wird die Auswahl übernommen.

Stylesheet

Das für das Element gültige Stylesheet. Wenn explizite kein Stylesheet ausgewählt worden ist, wird das erste Default-Stylesheet genutzt. Von den Stylesheet-Definitionen sind für Infopunkte die Bereiche **Schriftarteinstellungen** und **Erweiterte Sprachunterstützung** in **Allgemeine Einstellungen** sowie die Einstellungen für **Infopunkt und Hyperlink** des Stylesheet-Editors wichtig.

Praxistip: *Da Infopunkt-Felder häufig mit weisser Schrift auf dunklem Hintergrund dargestellt werden, kann die Lesbarkeit des Texts (Eigenschaft* **Wert***) mit Wahl eines Stylesheets mit fetter Schrift verbessert werden (z.B.* **Arial 10 Bold** *anstelle von* **Arial 10***).*

Tabulator-Reihenfolge

Ordnungsnummer des Elements in der Tabulator-Reihenfolge. Dieser Wert ist immer -1 und kann nicht verändert werden. Das bedeutet, dass das Element nicht Bestandteil der Tabulatoren-Reihenfolge ist.

druckbar

Diese Eigenschaft ist bei Infopunkt-Feldern immer ausgeschaltet, und kann nicht geändert werden.

sichtbar

Wenn das Kontrollkästchen gewählt ist, wird der Infopunkt auf dem Bildschirm angezeigt.

erforderlich

Diese Eigenschaft ist bei Infopunkt-Feldern immer ausgeschaltet, und kann nicht geändert werden.

Wert

In dieser Eigenschaft des Infopunkt-Felds wird der Text eingegeben, welcher im Feld erscheint und dieses als Infopunkt kennzeichnet. Typische Werte sind **?** oder **...** oder **i**. Mit Anklicken dieser Eigenschaft öffnet sich der Formeleditor zur Texteingabe.

Als Text im Infopunkt-Feld kann nur unformatierter Text eingegeben werden; HTML-Code wird nicht interpretiert und wörtlich wiedergegeben.

Mit einer Formel der Art

```
= "?|http://www.snapform.com"
```

kann ohne Festlegung eines Eintrags in der Eigenschaft **Textdetails** eine fertige HTML-Seite angezeigt werden. Diese Seite muss jedoch reinen HTML 3.2-kompatiblen Code aufweisen, ansonsten werden gewisse unerwünschte Texte angezeigt.

Gesperrt

Wenn das Kontrollkästchen gewählt ist, wird das Infopunkt-Feld im Snapform Designer blockiert, und kann nicht (versehentlich) verschoben werden. Zudem werden sämtliche übrigen Eigenschaften im Eigenschaften-Dialog deaktiviert. Der Zugriff auf die Eigenschaften ist erst wieder möglich, wenn dieses Kontrollkästchen deaktiviert wird.

Links

X-Koordinate der linken oberen Ecke der Umrandungsbox des Elements in Points. Diese Eigenschaft erlaubt eine genaue numerische Positionierung des Infopunkt-Felds.

Oben

Y-Koordinate der linken oberen Ecke der Umrandungsbox des Elements in Points. Diese Eigenschaft erlaubt eine genaue numerische Positionierung des Infopunkt-Felds.

Breite

Breite der Umrandungsbox des Infopunkt-Felds.

Höhe

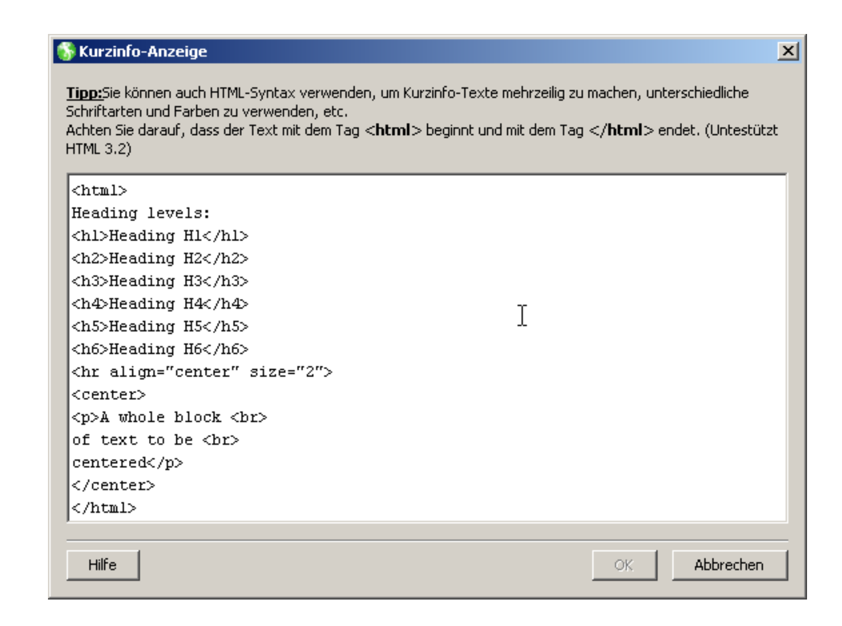
Höhe der Umrandungsbox des Infopunkt-Felds.

Textdetails

In dieser Eigenschaft wird der Text festgelegt, welcher beim Überfahren des Infopunkts mit der Maus angezeigt wird. Dieser Text kann entweder einfacher unformatierter Text sein, oder es kann formatierter Text in Form von HTML-Code eingegeben werden.

Wenn diese Eigenschaft gewählt wird, erscheint eine Schaltfläche **...** am rechten Rand des Eigenschaftsfelds. Mit Anklicken dieser Schaltfläche bzw. mit Doppelklick auf das restliche Eigenschaftsfeld öffnet sich das Info-Eingabefenster (siehe Fig. 4-159). Dieses Fenster ist identisch mit dem Kurzinfo-Anzeigefenster von Eingabefelder (siehe auch Abschnitt 4.4.2.1). Wie der Erläuterungstext angibt, kann hier HTML-Code eingegeben werden.

Fig. 4-159 *Info-Eingabe-Fenster 106*



Der in Fig. 4-159 als HTML definierte Text erscheint im Snapform Viewer wie in Fig. 4-160 dargestellt.

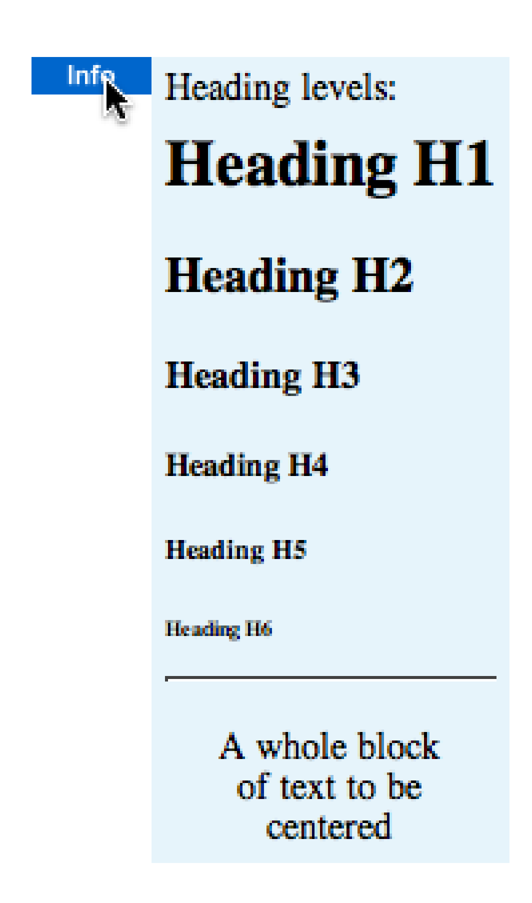


Fig. 4-160 *Infopunkt mit Text definiert gemäss Fig. 4-159*

Einfacher Text wird so dargestellt, wie er eingegeben worden ist, einschliesslich Zeilenumbrüche. Die Schriftart und -grösse richtet sich nach dem gewählten Stylesheet.

HTML-codierter Text wird entsprechend den Tags angezeigt. Die Breite des angezeigten Textbereichs richtet sich entweder nach den Angaben im HTML-Code oder, sofern diese nicht vorhanden sind, nach der Länge des längsten resultierenden Texts. Damit der Bereich nicht zu breit wird, empfiehlt es sich, die Breite festzulegen (indem der ganze Text in ein Tabellenfeld gesetzt wird und das Tabellenfeld eine definierte Breite hat). Alternativ kann mit dem
 Tag ein Zeilenumbruch forciert werden.

Die Grösse des angezeigten Infotexts ist vom Zoomfaktor des Viewers unabhängig. Das bedeutet, dass er bei kleinen Faktoren relativ zum Dokument grösser ist, und bei hohen Faktoren relativ kleiner. Dies ist bei der Ausgestaltung des Texts zu berücksichtigen.

Der unterstützte HTML-Code entspricht HTML 3.2 mit ein paar Ergänzungen von HTML 4.0.1. Eine Übersicht über die unterstützten Tags befindet sich in Abschnitt 4.3.6.2.

Hinweis: *Es ist möglich, referenzierte Bilder im Infotext einzufügen. Dabei muss jedoch beachtet werden, dass bei Benutzung des Formulars eine Internet-Verbindung stehen muss, damit die Bilder angezeigt werden können.*

Hinweis: *Wenn HTML-Code eingegeben wird, ist es wichtig, dass nach dem </HTML> Tag keine weiteren Zeichen (auch keine unsichtbaren Zeichen wie Leerzeichen oder Neue Zeile) vorhanden sind. Ist dies der Fall, wird der HTML-Code wörtlich angezeigt.*

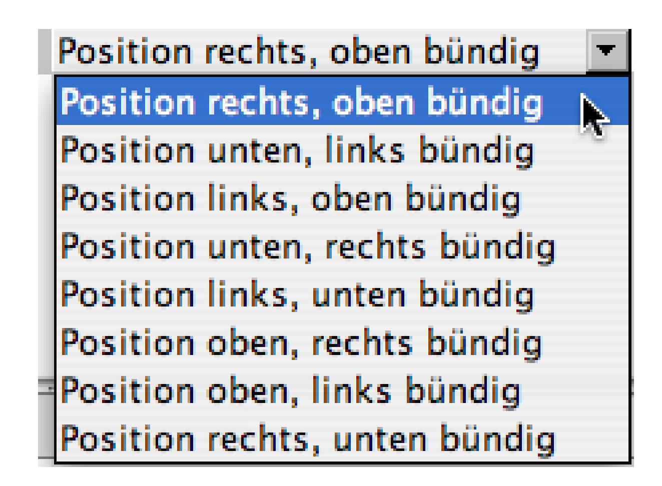
Hinweis: *Es kann sein, dass beim Bestätigen der Eingabe des Infotexts eine Warnung erscheint, dass Tags nicht korrekt seien. Dies ist auf einen etwas zu strikten Syntax-Checker von Snapform Designer zurückzuführen. Es wird geraten, die Änderungen mit* **Ja** *zu übernehmen und dann in der Vorschau zu kontrollieren, ob der HTML-Code trotz alledem korrekt interpretiert wird. Insbesondere bei Listen und Tabellen kann diese Meldung auftreten.*

Textposition

Diese Eigenschaft bestimmt, wie der Infotext in relation zum Infopunkt-Feld angezeigt wird. Dies erlaubt eine genaue Kontrolle darüber, welche Teile des Dokuments durch den erscheinenden Infotext abgedeckt werden.

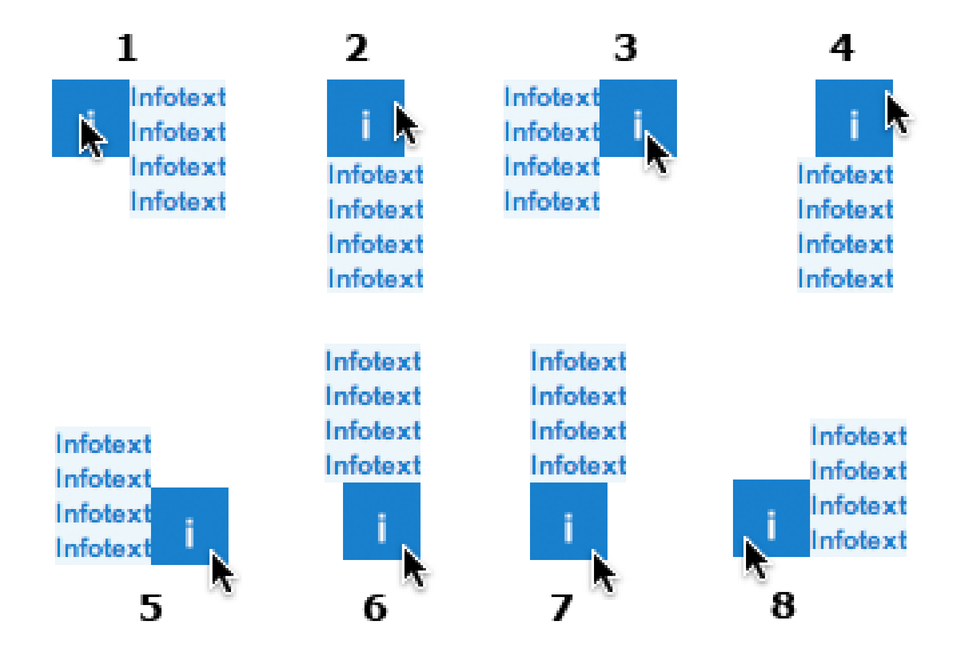
Wenn der Eigenschaftswert angeklickt wird, erscheint ein Drop Down Menu mit der in Fig. 4-161 gezeigten Auswahl.

Fig. 4-161 *Drop Down Menu mit Textpositionen für das Infopunkt-Feld*



Position legt fest, an welche Kante des Infopunkt-Felds der Infotext angesetzt wird. Bündig legt fest, auf welche Kante des Infopunkt-Felds der Infotext ausgerichtet wird. In Fig. 4-162 sind die acht Optionen bildlich dargestellt.

Fig. 4-162 *Darstellung der Textpositionen von Infopunkt-Feldern*



1 Position rechts, oben bündig
2 Position unten, links bündig
3 Position links, oben bündig
4 Position unten, rechts bündig
5 Position links, unten bündig

6 Position oben, rechts bündig
7 Position oben, links bündig
8 Position rechts, unten bündig

Hinweis: *Der Infotext wird rechnerisch über den Formularrand hinaus angezeigt. Ausserhalb des Dokuments liegende Teile werden abgeschnitten. Es wird daher angeraten, den Infotext immer in der Vorschau von Snapform Designer zu kontrollieren, und dabei auch verschiedene anzunehmende Zoom-Faktoren zu testen.*

Im Dokument infopoints.qdf (siehe Fig. 4-163 (Entwurfsmodus)) bzw. Fig. 4-164 (Ansicht in Snapform Viewer)) sind verschiedene Beispiele von Infopunkten gezeigt. In der folgenden Abbildung (siehe Fig. 4-165) sind die aktivierten Infotexte dargestellt (Das Bild ist mit Bildverarbeitung aus mehreren Teilbildern zusammengesetzt; es ist nicht möglich, mehrere Infotexte gleichzeitig zu öffnen).

Nach dem Selektieren der Infopunkt-Felder im Snapform Designer gibt das Eigenschaften-Fenster Auskunft über dessen jeweiligen Merkmale. Ein Doppelklick auf das Feld öffnet den Formeleditor zur Anzeige des Texts und ein Doppelklick auf das Eigenschaftsfeld **Textdetails** zeigt den Infotext an.

Fig. 4-163 *Infopunkt-Beispiele von infopoint.qdf, im Entwurfs-Modus 107*

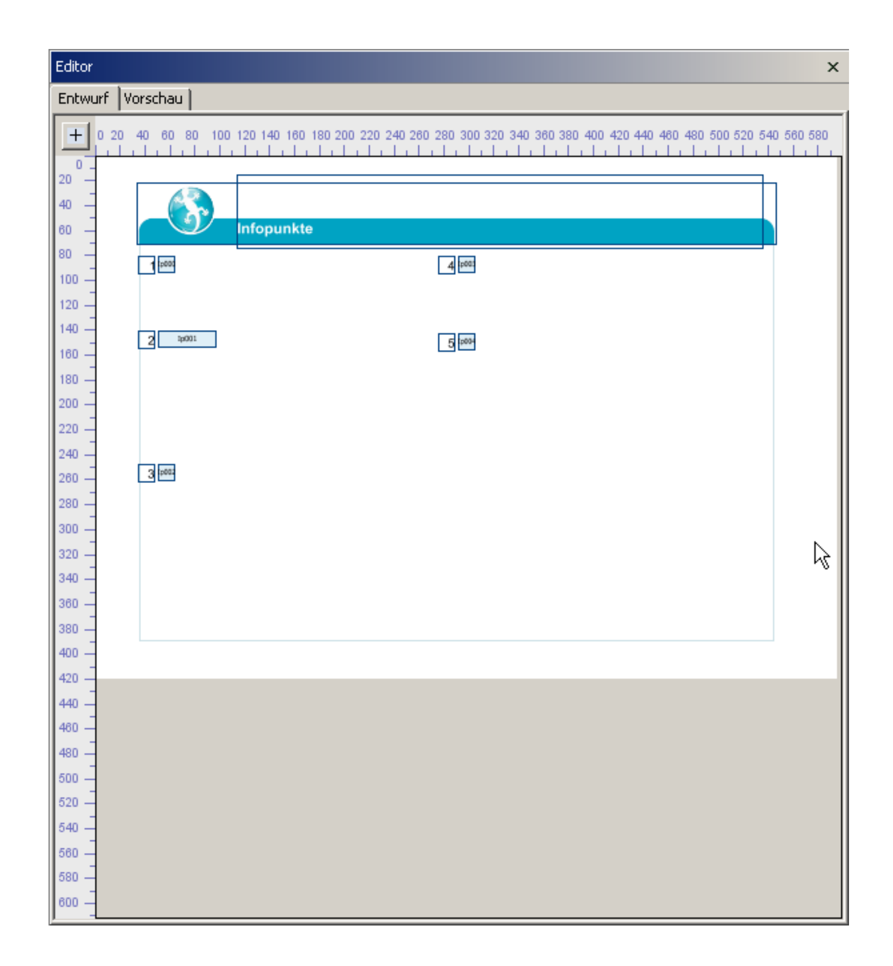
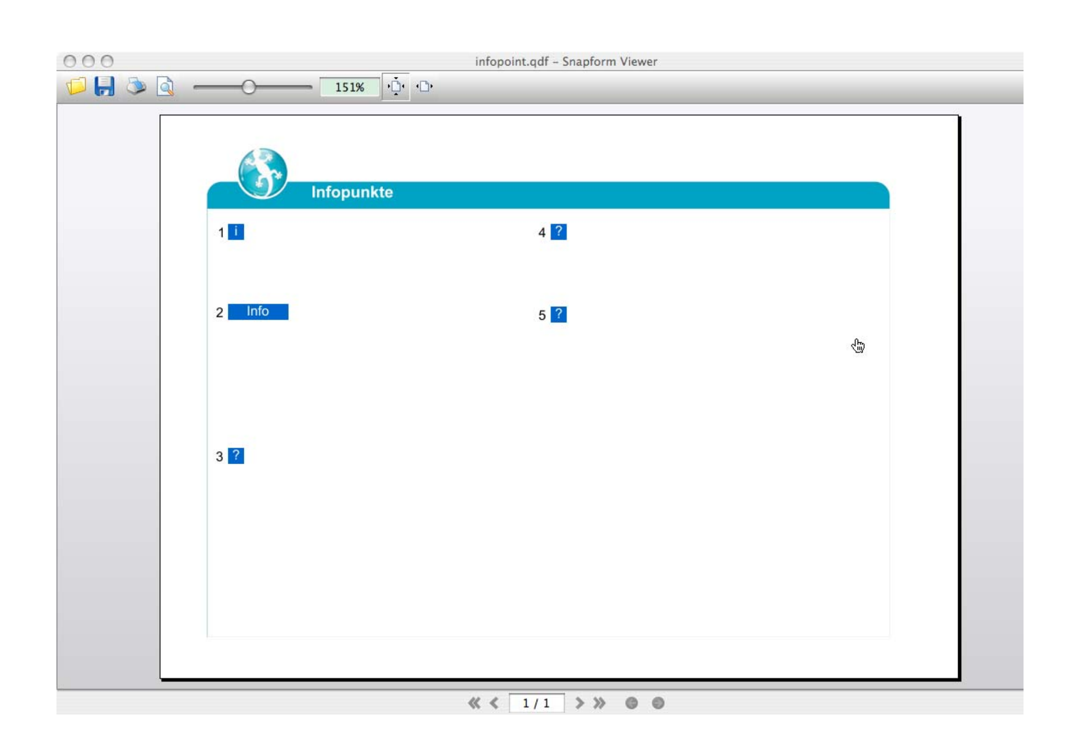


Fig. 4-164 *Infopunkt-Beispiele von infopoint.qdf in Snapform Viewer 108*



1 Ip000: Infopunkt, unformatierter Text, Wert von **Textdetails**:

```
Ein einfacher Infotext
```

2 Ip001: Infopunkt, unformatierter Text, Wert von **Textdetails**:

```
Ein einfacher, mehrzeiliger Text:

Snapform Infos:
infor@snapform.com
http://www.snapform.com
```

3 Ip002: Infopunkt, HTML-codierter Text mit In-line-Tags, Wert von **Textdetails**:

```
<html>
<p>Ein einfacher <em>HTML-codierter</em> Hilfetext.<br>
mehrzeilig mit <b>fetter</b>, <i>kursiver</i> und<br>
sogar <i><u>kursiv unterstrichener</u></i> Schrift.</p>
</html>
```

4 Ip003: Infopunkt, HTML-codierter Text mit verschiedenen Tags, Wert von **Textdetails**:

```
<html>
Heading levels:
<h1>Heading H1</h1>
<h2>Heading H2</h2>
<h3>Heading H3</h3>
<h4>Heading H4</h4>
<h5>Heading H5</h5>
<h6>Heading H6</h6>
<hr align="center" size="2">
<center>
```

```
<p>A whole block <br>
of text to be <br>
centered</p>
</center>
</html>
```

5. Ip004: Infopunkt, HTML-codierter Text mit Links und Bild, Wert von **Textdetails**:

```
<HTML>
<P>etwas komplexerer HTML-Text <BR>
Snapform Info sind erhältlich von<BR>
via E-Mail: <a href=
"mailto:info@snapform.com">info@snapform.com</a><BR>
Web: <a href=
"http://www.snapform.com">www.snapform.com</A><BR>
und hier der sch&ouml;ne Gecko:<BR>
<img src="http://www.snapform.com/img/s-blue.jpg"
alt="a non loadable image"></P>
</HTML>
```

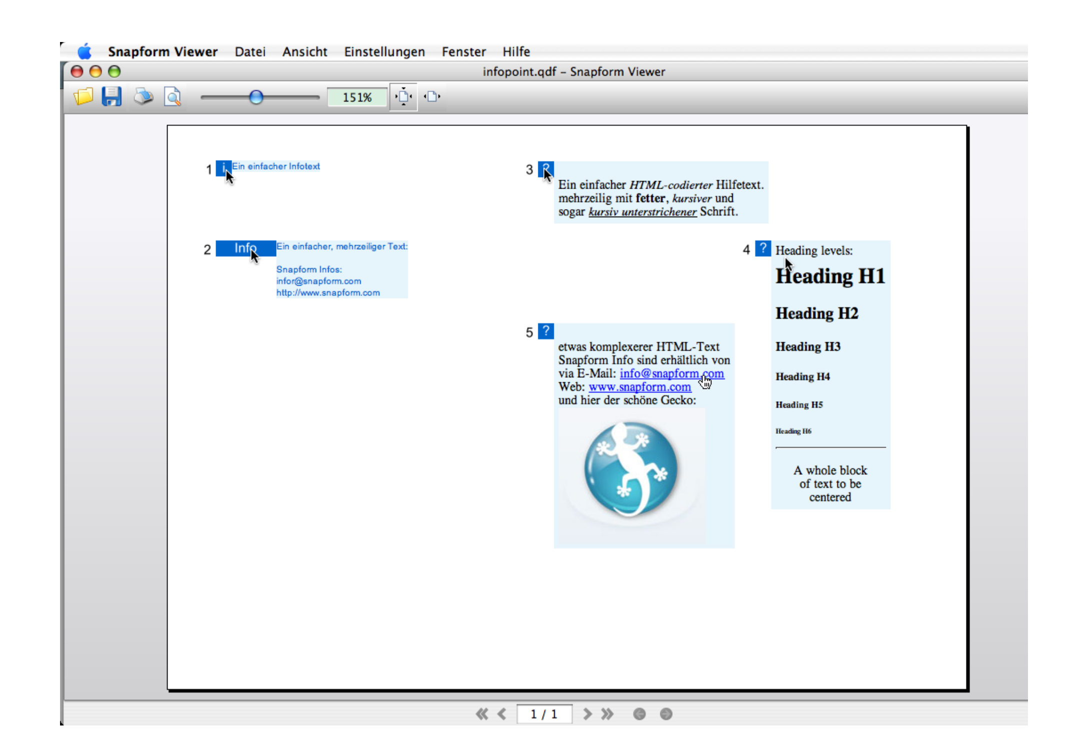


Fig. 4-165 *Infopunkt-Beispiele von infopoint.qdf in Snapform Viewer mit geöffneten Infotexten*

4.4.8.3 Schaltflächen

Ein weiteres aktives Element sind Schaltflächen. Dies sind Elemente, welche der Benutzer aktiviert (zum Beispiel durch Anklicken), und dadurch wird eine Aktion ausgelöst.

Schaltflächen im Sinn wie gewisse andere Formularsysteme besitzen, kennt Snapform nicht. Anstelle dessen ist ein Mechanismus vorhanden, welcher die weitaus am häufigsten Aktionen, welche Schaltflächen zugewiesen werden, über entsprechende Marken (Flags) im Snapform Viewer aktiviert. Man schätzt, dass damit, zusammen mit internen Hyperlinks) etwa 97% aller Schaltflächen-Aktionen abgedeckt sind.

Diese Formular-Aktionen werden in den Sicherheitseinstellungen des Formulars eingestellt (siehe Abschnitt 4.8.3.3).

4.4.8.4 Angewandte Aktive Elemente: ein Hilfesystem

Eine Anwendung von Aktiven Elementen ist ein Hilfesystem für ein Formular. Hilfesysteme werden häufig etwas vernachlässigt, sie lohnen sich jedoch sehr schnell, insbesondere bei eher komplexeren, oder aber sehr häufig von (für die Formularthematik) Laien benutzten Formularen. Durch den reduzierten Benutzersupportaufwand lohnt sich der Aufwand zum Aufbau eines Hilfesystems sehr schnell.

Beim Aufbau eines Hilfesystems ist zu beachten, dass es zwar für den ungeübten Benutzer gedacht ist (insbesondere die höheren Stufen), aber dass es einen erfahrenen Benutzer nicht behindern darf. Das bedeutet, dass jegliche Hilfestellung erst aufgrund einer Aktion des Benutzers angezeigt werden sollte. Ein Übermass an nicht verlangter Hilfe wird sehr schnell als Störung empfunden.

Ein Hilfesystem umfasst mehrere Stufen:

1. direkte Hinweise auf Feldebene

   Die 1. Stufe des Hilfesystems sollte besagen, was in das zugehörige Feld eingegeben werden soll. Dies ist eine Ergänzung zur Bezeichnung des Felds auf dem Formular. Die 1. Stufe soll dem mit der Materie vertrauten Benutzer durch das Formular helfen. Es ist jedoch nicht die Aufgabe der 1. Stufe, zu erklären, warum diese Eingabe zu erfolgen hat, und welches ihre Bedeutung ist.

Die Information der 1. Stufe wird üblicherweise auch von Screenreadern verwendet, um den Benutzer zu führen. Dies ist beim Verfassen des Texts zu berücksichtigen.

In Snapform wird die 1. Stufe über die Kurz-Info von Eingabefeldern realisiert (siehe Abschnitt 4.4.2.1).

2. thematische Hinweise auf Feldgruppenebene

   Die 2. Stufe des Hilfesystems bezieht sich auf die Erläuterung von logisch zusammenhängenden Bereichen des Formulars. Hier wird auf die Bedeutung des Bereichs eingegangen. Diese Informationen sind fachbezogen, und helfen dem Neuling in der Materie, sich zurechtzufinden. Die 2. Stufe sollte sich jedoch nicht mit eingabespezifischen Dingen befassen (das ist etwas für die 1. Stufe), es sei denn, diese Eingaben stünden im grösseren Zusammenhang.

   In Snapform wird die 2. Stufe über Infopunkte realisiert (siehe Abschnitt 4.4.8.2).

3. vertiefte thematische Hinweise auf Formularbereichsebene

   Die 3. Stufe des Hilfesystems bezieht sich auf weitergehende Zusammenhänge im Formular. Diese Informationen sind nach wie vor im Dokument enthalten, werden aber seltener gebraucht. Der Zugriff auf diese Informationen kann nicht mehr direkt erfolgen, sondern über einen "Hauptschalter", welcher quasi eine "Hilfeebene" ein- und ausblendet.

   In Snapform wird die 3. Stufe über Infopunkte entweder über eine zusätzliche Seite, angesprungen via Hyperlink oder mittels einer Kategorie, welche über ein Kontrollkästchen gesteuert ein- und ausgeschaltet wird, realisiert. der eigentliche Zugriff auf die Informationen erfolgt über Infopunkte.

4. Hintergrundinformationen auf Formularebene.

   Die 4. Stufe des Hilfesystems behandelt grundsätzliche Hintergrund-Themen zum Formular, wie gesetzliche Grundlagen

etc. Diese Informationen sind nicht mehr Bestandteil des Formulars. Beim Nutzen dieser Informationen wird das Formular selbst verlassen. Diese Stufe wird nur dann benutzt, wenn ein vertieftes Interesse vorhanden ist, oder wenn es darum geht, Sonderfälle zu behandeln.

Elemente der 4 Stufe können entweder offen und jederzeit zugänglich, oder aber versteckt, über die 3. Stufe, eventuell auch 2. Stufe, sichtbar gemacht werden.

In Snapform wird die 4. Stufe über Hyperlinks (siehe Abschnitt 4.4.8.1) oder über in Infotexte eingebettete Links (HTML-codierte Texte, siehe Abschnitt 4.4.8.2) realisiert.

4.5 Aktionen und Logik

Der grösste Vorteil von elektronischen Formularen gegenüber Papier-basierten Formularen ist die Möglichkeit, "Intelligenz" einzubauen, und das Formular interaktiv zu machen. Dies erfolgt normalerweise mit einer Formel- oder Programmiersprache. In Snapform gelangt eine Formelsprache zur Anwendung, welche mit derjenigen von Tabellenkalkulationen verwandt ist.

Der Unterschied zwischen Formel- und Programmiersprachen zeigt sich im Umfang von Kontrollstrukturen und Funktionen. Dieser Umfang ist bei Formelsprachen eher geringer. Andererseits kann eine Formel mit einen übersichtlichen Codegenerator erzeugt werden, was die Benutzung wesentlich einfacher macht.

Ein weiterer Unterschied ist der, dass bei Formeln feldbezogene Resultate "hereingezogen" werden, währenddem in Programmiersprachen feldbezogene Resultate auch in das Feld "eingespiesen" werden können. Die Folge ist, dass Berechnungsformeln demjenigen Feld zugeordnet werden müssen, welches das Resultat anzeigt.

Der Formeleditor mit seinen Benutzungselementen ist in Abschnitt 3.2.8 beschrieben.

Formeln werden entweder über eine Aktion (der Benutzer "macht etwas") oder über Logik (das Formular "macht etwas") abgearbeitet.

4.5.1 Träger von Aktionen und Logik

Aktionen und Logik benötigen einen Träger, damit sie ausgeführt werden können. Als Träger von Formeln kommen Felder und Funktionen in Frage. In älteren Snapform-Versionen konnten Formeln einzig an Beschreibungsfelder angeknüpft werden. Dies ist nicht mehr notwendig, und Formeln können nun jedem Feldtyp angefügt werden.

Zudem besteht die Möglichkeit, Formeln als Funktionen zu definieren, was bei mehrfacher Verwendung grosse Vorteile bietet.

4.5.1.1 Felder

Formeln in Feldern sind im Wesentlichen “auf sich selbst bezogen”. Das Resultat ist entweder gänzlich unabhängig von Feldern (zum Beispiel ein- oder ausblenden einer Seite), oder es wird direkt in das aufrufende Feld geschrieben.

Formeln werden in der Eigenschaft Wert des betreffenden Felds eingegeben. Berechnungsformeln beginnen immer mit einem Gleichheitszeichen.

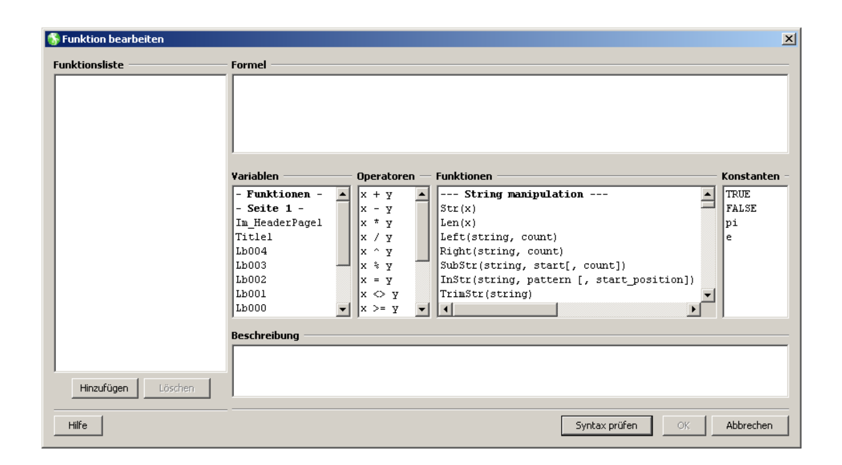
4.5.1.2 Funktionen

Es kann vorkommen, dass Formeln (oder Komponenten von Formeln) in mehreren Feldern eingesetzt sind. In diesem Fall lohnt es sich, Funktionen zu schaffen, welche von den Formeln dieser Felder aufgerufen werden können. Damit wird der Aufwand zum Testen und Warten der Formeln stark reduziert, und die Fehlerquelle bei Wiederholung des gleichen Codes wird unterdrückt.

Funktionen sind genau gleich aufgebaut wie Formeln, und können entsprechend benutzt werden. Im Prinzip können Funktionen als “Abkürzungen” für Formelbestandteile betrachtet werden.

Funktionen werden im Funktioneneditor (siehe Fig. 4-166) aufgebaut.

Fig. 4-166 *Der Funktionseditor 109*



Der Funktioneneditor ist in der Benutzung identisch wie der Formeleditor. Der Hauptunterschied ist der, dass ein zusätzliches Fenster **Funktionsliste** vorhanden ist, in welchem die Funktionsnamen erscheinen.

Mit Anklicken von **Hinzufügen** wird eine Funktionsbezeichnung in die Liste hinzugefügt. Da diese Funktionsbezeichnung nicht allzu selbsterklärend ist, sollte sie dem Anwendungszweck gemäss geändert werden. Bearbeitet wird die Funktion, wenn sie in der **Funktionsliste** selektiert ist.

Mit Anklicken von **Löschen** kann die selektierte Funktion entfernt werden. Es muss dabei beachtet werden, dass diese Funktion dann nicht mehr zur Verfügung steht, und dass unter Umständen Formeln im Formular nicht mehr korrekt funktionieren.

4.5.2 In Snapform verfügbare Aktionen und Logik

Die verfügbaren Funktionen sind gruppiert nach ihrem Einsatzzweck. Die nachstehenden Abschnitte geben die Gruppierung der Funktionen im Formeleditor wieder (das Kapitel 5, "Formelsprache-Referenz" führt die Funktionen in alphabetischer Folge auf).

4.5.2.1 String-Funktionen

String-Funktionen beziehen sich auf Strings (Zeichenketten). Grundsätzlich lässt sich jedes Element in einen String umwandeln, so dass diese Funktionen einen weit gehenden Wirkungsbereich haben.

Die folgenden Funktionen sind den String-Funktionen zugeordnet:

Str(x)

Wandelt den Eingabewert in einen String um.

Beispiel siehe func_string1.qdf

Len(x)

Gibt die Länge (Anzahl Zeichen) eines Strings zurück.

Beispiel siehe func_string1.qdf

Left(string, count)

Gibt die ersten count Zeichen, von links her gezählt, des Strings string zurück.

Beispiel siehe func_string1.qdf

Right(string, count)

Gibt die ersten count Zeichen, von rechts her gezählt, des Strings string zurück.

Beispiel siehe func_string1.qdf

SubStr(string, start[, count])

Bildet einen Teilstring ab Zeichenposition start mit der Länge count.

Beispiel siehe func_string1.qdf

InStr(string, pattern [, start_position])

Gibt die Position von pattern im string, beginnend mit der Suche ab Position start_position zurück.

Beispiel siehe func_string2.qdf

TrimStr(string)

Entfernt vor- und nachlaufende Leerzeichen von einem String.

Beispiel siehe func_string2.qdf

SubArray(byte_array, start [, count])

Bildet eine Sub-Folge einer Bytefolge byte_array, beginnend mit Byte Nummer start und der Anzahl count Bytes.

Beispiel siehe func_string2.qdf

getBytes(string)

Codiert die eingegebene Zeichenkette zu einer Bytefolge, welche in einem 2D-Barcode mit Codierung Binär eingegeben werden kann.

Beispiel siehe func_string2.qdf

4.5.2.2 Regular Expression-Funktionen

Regular Expression-Funktionen beziehen sind eine Erweiterung von String-Funktionen, und dienen zum Analysieren von Strings gegenüber einem durch einen Regular Expression definierten Muster.

Die folgenden Funktionen sind den Regular Expression-Funktionen zugeordnet:

find(string, regex[, start])

Findet die Position der ersten nach Position start vorkommenden Fundstelle des in regex definierten Musters im Eingabewert string.

Beispiel siehe func_regex.qdf

match(string, regex)

Stellt fest, ob das in regex definierte Muster im Eingabewert string enthalten ist.

Beispiel siehe func_regex.qdf

replaceAll(orig_string , regex, replace_with)

Ersetzt sämtliche Fundstellen des in regex definierten Musters im Eingabewert orig_string durch den String replace_with.

Beispiel siehe func_regex.qdf

4.5.2.3 Mathematik-Funktionen

Mathematik-Funktionen werden auf Zahlen angewandt und führen mathematische Operationen aus.

Die folgenden Funktionen sind den Mathematik-Funktionen zugeordnet

Abs(x)

Berechnet den Absolut-Wert des Eingabewerts.

Beispiel siehe func_math1.qdf

Max(x, y)

Bestimmt den grösseren der beiden Werte.

Beispiel siehe func_math1.qdf

Min(x, y)

Bestimmt den kleineren der beiden Werte.

Beispiel siehe func_math1.qdf

Sin(x)

Berechnet den Sinus eines in Grad definierten Winkels.

Beispiel siehe func_math2.qdf

Cos(x)

Berechnet den Cosinus eines in Grad definierten Winkels.

Beispiel siehe func_math2.qdf

Tan(x)

Berechnet den Tangens eines in Grad definierten Winkels.

Beispiel siehe func_math2.qdf

Asin(x)

Gibt den Arcus Sinus des Eingabewerts in Grad zurück.

Beispiel siehe func_math2.qdf

Acos(x)

Gibt den Arcus Cosinus des Eingabewerts in Grad zurück.

Beispiel siehe func_math2.qdf

Atan(x)

Gibt den Arcus Tangens des Eingabewerts in Grad zurück.

Beispiel siehe func_math2.qdf

Round(x)

Rundet den Eingabewert auf die nächste ganze Zahl.

Beispiel siehe func_math1.qdf

Ceil(x)

Rundet den Eingabewert auf die nächstgrössere ganze Zahl auf.

Beispiel siehe func_math1.qdf

Floor(x)

Rundet den Eingabewert auf die nächstkleinere ganze Zahl ab.

Beispiel siehe func_math1.qdf

Trunc(x)

Rundet den Absolutwert des Eingabewerts auf die nächstkleinere ganze Zahl ab.

Beispiel siehe func_math1.qdf

Cover(x)

Rundet den Absolutwert des Eingabewerts auf die nächstgrössere ganze Zahl auf.

Beispiel siehe func_math1.qdf

Exp(x)

Berechnet den Wert der Exponentialfunktion ex.

Beispiel siehe func_math2.qdf

Log(x)

Berechnet den natürlichen Logarithmus von x (ln x).

Beispiel siehe func_math2.qdf

Sqrt(x)

Berechnet die Quadratwurzel des Eingabewerts.

Beispiel siehe func_math2.qdf

IncVersion(initial_value, step)

Schleifenzähler: setzt einen Zähler auf inital_value und erhöht ihn bei jedem folgenden Aufruf um step.

Beispiel siehe func_math1.qdf

4.5.2.4 Tabellen-Funktionen

Tabellen-Funktionen werde auf Tabellen angewandt und dienen hauptsächlich zum Zugriff auf die Tabellendaten.

Die folgenden Funktionen sind den Tabellenfunktionen zugeordnet:

Total(table, column)

Summiert die Werte der Spalte column zu Summe.

Beispiel siehe func_table1.qdf

StrTotal(table, column)

Fügt die Tabellenwerte in Spalte column zu einem String zusammen.

Beispiel siehe func_table1.qdf

TimeTotal(table, column)

Summiert die Werte der Spalte column zu einer Uhrzeit.

Beispiel siehe func_table2.qdf

Cell(table, column, row)

Stellt den Wert der Tabellenzelle von Tabelle table, Spalte column, Zeile row dar.

Beispiel siehe func_table2.qdf

TblValues(table, tbl_start, tbl_end, row_start, row_end, delimiter, column [, column[, …]])

Erzeugt einen String mit zeilenweise zusammengefügten Tabellenwerten und spezifizierten Delimitern.

Beispiel siehe func_table1.qdf

getValues(delimiter, value1 [, value2[, ...]])

Fügt die Werte der angegebenen Felder zu einem String zusammen und benutzt dabei das Trennzeichen delimiter.

Beispiel siehe func_table2.qdf

getRow()

Gibt die Zeilennummer des betreffenden Felds in der zugehörigen Tabelle an.

Beispiel siehe func_table2.qdf

getCol()

Gibt die Spaltennummer des betreffenden Felds in der zugehörigen Tabelle an.

Beispiel siehe func_table2.qdf

4.5.2.5 Datums-Funktionen

Datums-Funktionen werden auf Datum und Uhrzeit sowie Zeitdifferenzen angewandt.

Die folgenden Funktionen sind den Datums-Funktionen zugeordnet:

Today()

Gibt das aktuelle Datum zurück.

Beispiel siehe func_date.qdf

Now()

Gibt die aktuelle Uhrzeit zurück.

Beispiel siehe func_time.qdf

Date(year, month, day)

Erzeugt ein Datumsobjekt aus den Argumenten.

Beispiel siehe func_date.qdf

Time(hour, minute[, second])

Erzeugt ein Uhrzeitobjekt aus den Argumenten.

Beispiel siehe func_time.qdf

getDay(date_value)

Gibt den Monatstag des Datums date_value an.

Beispiel siehe func_date.qdf

getMonth(date_value)

Gibt den Monatswert des Datums date_value an.

Beispiel siehe func_date.qdf

getYear(date_value)

Gibt die Jahrzahl des Datums date_value an.

Beispiel siehe func_date.qdf

getHour(time_value)

Gibt den Stundenwert der Uhrzeit time_value an.

Beispiel siehe func_time.qdf

getMinute(time_value)

Gibt den Minutenwert der Uhrzeit time_value an.

Beispiel siehe func_time.qdf

addDay(date_value, number)

Berechnet das Datum, welches gegeüber dem Datum date_value um number Tage entfernt ist.

Beispiel siehe func_date.qdf

addMonth(date_value, number)

Berechnet das Datum, welches gegeüber dem Datum date_value um number Monate entfernt ist.

Beispiel siehe func_date.qdf

addYear(date_value, number)

Berechnet das Datum, welches gegeüber dem Datum date_value um number Jahre entfernt ist.

Beispiel siehe func_date.qdf

addHour(time_value, number)

Berechnet die Uhrzeit, welche gegeüber der Uhrzeit time_value um number Stunden entfernt ist.

Beispiel siehe func_time.qdf

addMinute(time_value, number)

Berechnet die Uhrzeit, welche gegeüber der Uhrzeit time_value um number Minuten entfernt ist.

Beispiel siehe func_time.qdf

calcDays(date_first, date_second)

Berechnet die Anzahl Tage zwischen den beiden Eingabedaten.

Beispiel siehe func_date.qdf

calcHours(date_first, date_second)

Berechnet die Anzahl Stunden zwischen den beiden Eingabedaten.

Beispiel siehe func_time.qdf

calcMinutes(date_first, date_second)

Berechnet die Anzahl Minuten zwischen den beiden Eingabedaten.

Beispiel siehe func_time.qdf

4.5.2.6 Status-Funktionen

Die Status-Funktionen dienen zum Setzen des Status (druckbar, sichtbar, schreibgeschützt) von Feldern, Seiten und Kategorien.

Die folgenden Funktionen sind den Status-Funktionen zugeordnet:

setVisible(category_list, true|false)

Setzt die Eigenschaft sichtbar der betreffenden Kategorie(n).

Beispiel siehe func_status.qdf

setPrintable(category_list, true|false)

Setzt die Eigenschaft druckbar der betreffenden Kategorie(n).

Beispiel siehe func_status.qdf

setWritable(category_list, true|false)

Setzt sinngemäss die Eigenschaft schreibgeschützt der betreffenden Kategorie(n).

Beispiel siehe func_status.qdf

setFieldVisible(true|false, field_name)

Setzt die Eigenschaft sichtbar des betreffenden Felds.

Beispiel siehe func_status.qdf

setFieldPrintable(true|false, field_name)

Setzt die Eigenschaft druckbar des betreffenden Felds.

Beispiel siehe func_status.qdf

setFieldWritable(true|false, field_name)

Setzt sinngemäss die Eigenschaft schreibgeschützt des betreffenden Felds.

Beispiel siehe func_status.qdf

setPageVisible(true|false, page_number [, page_number[, ...]])

Setzt die Eigenschaft sichtbar der betreffenden Seite(n).

Beispiel siehe func_status.qdf

setPagePrintable(true|false, page_number [, page_number[,...]])

Setzt die Eigenschaft druckbar der betreffenden Seite(n).

Beispiel siehe func_status.qdf

4.5.2.7 Konsistenzprüfungs-Funktionen

Konsistenzprüfungs-Funktionen dienen zur Erzeugung und Verifizierung von Prüfsummen über Eingabedaten.

Die folgenden Funktionen sind den Konsistenzprüfung-Funktionen zugeordnet:

checkCardNumber(value)

Validiert Kreditkartennummern.

Beispiel siehe func_consist.qdf

getMod10(value)

Gibt eine mit Hilfe des "MOD-10-Algorithmus" erzeugte Prüfsumme über den Eingabwert value zurück.

Beispiel siehe func_consist.qdf

checkMod10(value)

Validiert Nummern unter Verwendung des "MOD-10-Algorithmus".

Beispiel siehe func_consist.qdf

getMod10r(value)

Gibt eine mit Hilfe des "rekursiven MOD-10-Algorithmus" erzeugte Prüfsumme über den Eingabwert value zurück.

Beispiel siehe func_consist.qdf

checkMod10r(value)

Validiert Nummern unter Verwendung des "rekursiven MOD-10-Algorithmus".

Beispiel siehe func_consist.qdf

4.5.2.8 Identifikations-Funktionen

Identifikations-Funktionen dienen zum Erzeugen von eindeutigen, einmaligen Zeichenfolgen, welche eine exakte Zuordnung zu Datensätzen erlauben.

Die folgenden Funktionen sind den Identifikations-Funktionen zugeordnet:

createGUID()

Erzeugt einen einmaligen Identifikationsstring (GUID, Global Unique Identifier).

Beispiel siehe func_ident.qdf

getSessionID()

Erzeugt einen einmaligen Identifikationsstring (GUID, Global Unique Identifier), welcher der laufenden Arbeitssitzung mit Snapform Viewer zugewiesen werden kann.

Beispiel siehe func_ident.qdf

getPrintID()

Erzeugt einen einmaligen Identifikationsstring (GUID, Global Unique Identifier), welcher mit einem Druckauftrag verknüpft werden kann.

Beispiel siehe func_ident.qdf

getDataID()

Erzeugt einen einmaligen Identifikationsstring (GUID, Global Unique Identifier), welcher mit einem Satz Formulardaten verknüpft werden kann.

Beispiel siehe func_ident.qdf

getRandomString(length [, digits])

Erzeugt einen zufällig aufgebauten String der Länge length.

Beispiel siehe func_ident.qdf

getPrintIdStr(length [, digits]

Erzeugt einen frei konfigurierbaren einmaligen Identifikationsstring, welcher mit einem Druckauftrag verknüpft werden kann.

Beispiel siehe func_ident.qdf

4.5.2.9 Combobox-Funktionen

Zu den Combobox-Funktionen gehören Funktionen, welche mit Combobox-Formularelementen zusammenhängen.

Die folgenden Funktionen sind den Combobox-Funktionen zugeordnet:

setList(ComboBox, str_items, str_values)

Konfiguriert die Auswahlliste für das Auswahllistenfeld ComboBox.

Beispiel siehe func_combobox.qdf

getSelectedIndex(ComboBox)

Gibt den Index der aktuellen Auswahl eines Auswahllistenfelds zurück.

Beispiel siehe func_combobox.qdf

setSelectedIndex(ComboBox, index)

Setzt die aktuelle Auswahl des Listenfelds ComboBox auf den Wert mit Index index.

Beispiel siehe func_combobox.qdf

4.5.2.10 Spezial-Funktionen

Zu den Spezial-Funktionen gehören Funktionen, welche nicht zu einer der oben stehenden Gruppen zugeordnet werden können.

Die folgenden Funktionen sind den Spezial-Funktionen zugeordnet:

Compress(x)

Erzeugt aus dem Eingabewert eine mit ZLIB komprimierte Bytefolge, welche in einem Barcode mit Codierungsmethode Binär dargestellt werden kann.

Beispiel siehe func_barcodes.qdf

if(logical_test, value_if_true [,value_if_false](

Führt eine Vergleichs-Operation auf den darin angegebenen Operanden durch und gibt je nach Resultat einen entsprechenden Wert zurück.

Beispiel siehe func_misc.qdf

isEmpty(x)

Prüft ob x der Leere Wert (bzw. null) ist.

Beispiel siehe func_misc.qdf

importData(url)

Importiert eine Datendatei von url.

Beispiel siehe func_misc.qdf

getVersion()

Gibt die Versionsnummer der benutzten Snapform-Applikation (Designer, Viewer) zurück.

Beispiel siehe func_ident.qdf

getContent(url)

Lädt den (Text)-Inhalt von url.

Beispiel siehe func_misc.qdf

4.6 Integrität des Formulars

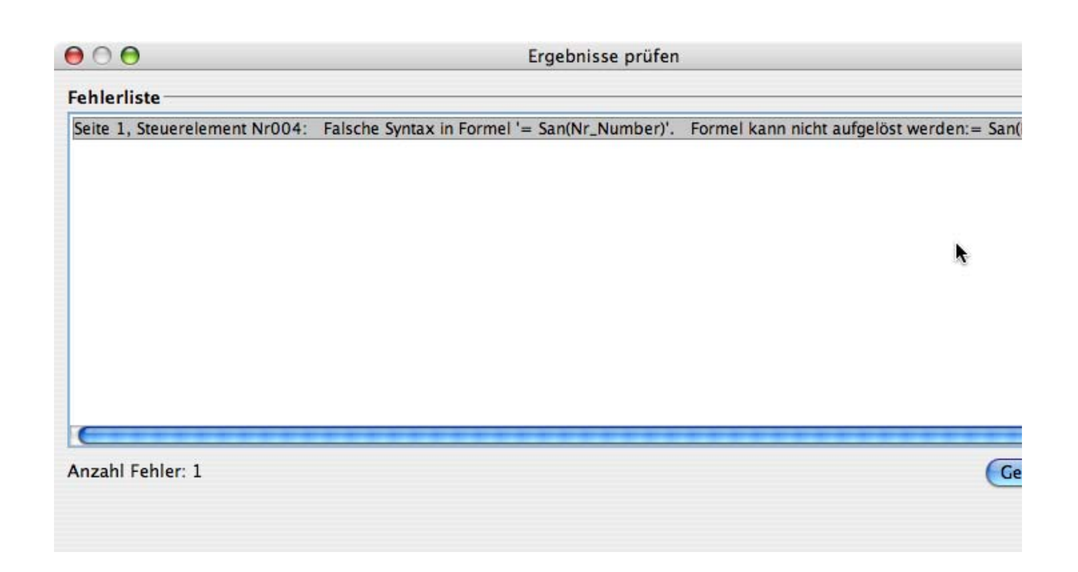
Einer der wichtigsten Schritte beim Fertigstellen eines Formulars ist das Sicherstellen der Integrität. Das bedeutet, dass das Formular funktionsfähig und vollständig ist.

Der erste Schritt ist das Prüfen des Formulars, um allfällige innere Fehler zu finden. Dies erfolgt mit der Funktion **Formular prüfen** (Werkzeugleiste oder Menu **Extras**).

Da es möglich ist, fehlerhafte Formeln zu übernehmen, ist diese Prüfung wichtig, da sie diese Fehler aufdeckt.

Wenn Fehler festgestellt werden, öffnet sich die Fehlerliste (siehe Fig. 4-167).

Fig. 4-167 *Fehlerliste von* **Formular prüfen**



In diesem Beispiel wurde ein Fehler festgestellt, auf Seite 1, im Zahlenfeld **Nr004**. Die Formel ist falsch, die Funktion heisst nicht **`San()`**, sondern **`Sin()`**. Die Zeile ist selektiert, und mit Anklicken von **Gehe zu Fehler** wird das betreffende Formularelement aktiv, und der Fehler kann behoben werden.

Wenn diese Funktion keine Fehler mehr anzeigt, bedeutet das, dass das Formular als solches funktioniert. Das heisst allerdings nicht, dass es

korrekt ist, da konzeptionelle Fehler mit dieser Funktion nicht gefunden werden können.

Der nächste Schritt in der Überprüfung des Formulars besteht nun aus Testen. Zuerst sollte mit "möglichen" Eingabewerten geprüft werden, ob die Resultate stimmen, aber die Kontrolle mit "unmöglichen" und "vorsätzlich falschen" Werten ist ebenso wichtig.

4.7 Stilvorlagen

Stilvorlagen steuern das Erscheinungsbild von Formularelementen. Dazu gehört Schriftart und -grösse, Farbe von Feldhintergrund und Umrandung in Abhängigkeit der Funktion sowie von Tabellen.

Stilvorlagen können exportiert und importiert werden, so dass es möglich ist, die Formulare mit einem konsistenten Erscheinungsbild aufzubauen.

Die Stilvorlagen sind mit dem jeweiligen Formular gespeichert. Jedes Formular trägt daher seine eigenen Stilvorlagen.

Bearbeitet werden die Stilvorlagen im Stilvorlageneditor.

4.7.1 Der Stilvorlageneditor

Der Stilvorlageneditor (siehe Fig. 4-168) wird mit der Funktion **Stylesheet-Editor** im Menu **Extras** gestartet.

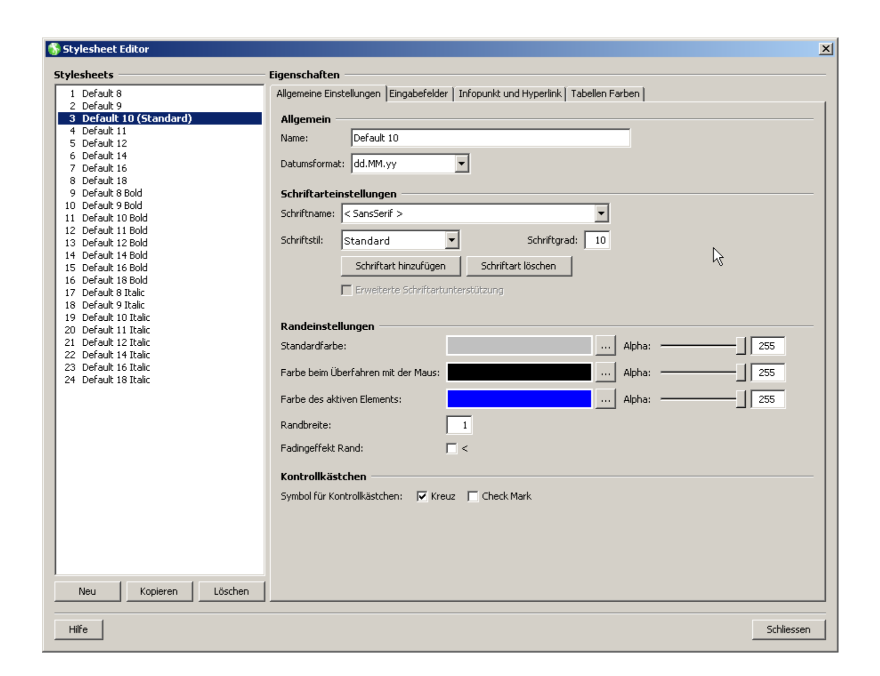


Fig. 4-168 *Der Stilvorlageneditor, Allgemeine Einstellungen 110*

Der Stilvorlageneditor umfasst für eine vollständig definierte Stilvorlage vier Einstellungsfenster. Die **Allgemeinen Einstellungen** umfassen Schriftarten, Grund-Datumsformat sowie Randeinstellungen für Felder.

Im Auswahlfeld auf der linken Seite sind sämtliche Stilvorlagen des aktuellen Dokuments aufgeführt. Die Vorlagen mit Schriftnamen **Arial** und **Courier** sind in der Grundeinstellung vorhanden, und bestehen im Wesentlichen für jedes Formular.

Mit der Schaltfläche **Neu** wird eine neue Stilvorlage hinzugefügt. Mit der Schaltfläche **Kopieren** wird von der aktuellen Stilvorlage eine Kopie erstellt und hinzugefügt, und mit der Schaltfläche **Löschen** wird die aktuelle Stilvorlage entfernt.

Im Feld **Name** ist der Name der Stilvorlage eingegeben. Unter diesem Namen erscheint die Stilvorlage auch in der Auswahlliste der Formularelement-Eigenschaft **Stilvorlage**.

Das Feld **Datumsformat** zeigt eine Auswahlliste für die Grundeinstellung des Datumsformats. Dieses kann bei Formularelementen mit einer Eigenschaft **Datumsformat** jedoch übersteuert werden. Das aus der Liste gewählte Format ist auch das Datumsformat für die Eingabe von Daten in das Feld.

Im Bereich **Schriftarteinstellungen** wird die Schriftart für das Formularelement gewählt. Das Auswahlfeld **Schriftname** hat eine Liste von verfügbaren Schriftarten. Das Auswahlfeld **Schriftstil** zeigt abhängig von der Schriftart die verfügbaren Stile (fett, kursiv, etc.) Im Feld **Schriftgrad** wird die Schriftgrösse festgelegt.

Mit der Schaltfläche **Schriftart hinzufügen** können weitere auf dem System installierte Schriften hinzugefügt werden. Es öffnet sich ein Datei Öffnen-Dialog, über welchen Schriftdateien gewählt werden können. Zur Zeit wird ausschliesslich das TrueType-Format (**.ttf**) unterstützt.

Hinweis: *Schriften kommen mit einer Vielzahl von Spielarten der Lizenzierung. Es ist in der Verantwortung des Benutzers von Snapform Designer, diese Lizenzbestimmungen entsprechend zu berücksichtigen.*

Mit der Schaltfläche **Schriftart löschen** kann eine Schriftart wieder entfernt werden.

Snapform bietet die Möglichkeit zur sogenannten erweiterten Sprachunterstützung, welche plattformspezifische Schriftarten zulässt. Wenn das Kontrollkästchen **Erweiterte Sprachunterstützung** eingeschaltet ist, werden die drei Eingabefelder **Schriftart unter Windows**, **Schriftart unter Mac OS X** und **Schriftart unter Unix** aktiv, und es kann die gewünschte Schriftart eingegeben werden.

Die im **Bereich Randeinstellungen** vorhandenen Optionen gelten für alle Formularelemente (soweit sie in den Eigenschaften eingeschaltet sind).

Die **Standardfarbe** ist die "Grundstellung" des Formularelements. Mit Anklicken der Schaltfläche **...** öffnet sich das Farbauswahlfenster (siehe Abschnitt 3.2.9.2), und die gewünschte Farbe kann eingestellt werden. Der Schieber **Alpha** steuert die Transparenz der Farbe. Je geringer der Wert ist, desto transparenter ist der Rand (oder die Fläche) Der Wert für **Alpha** wird im Feld neben dem Schieber angezeigt, und kann auch direkt eingegeben werden. Der Wert kann zwischen **0** und **255** liegen.

Hinweis: *Die Prinzipien für die Farbeinstellungen gelten für alle Bereiche im Stilvorlageneditor.*

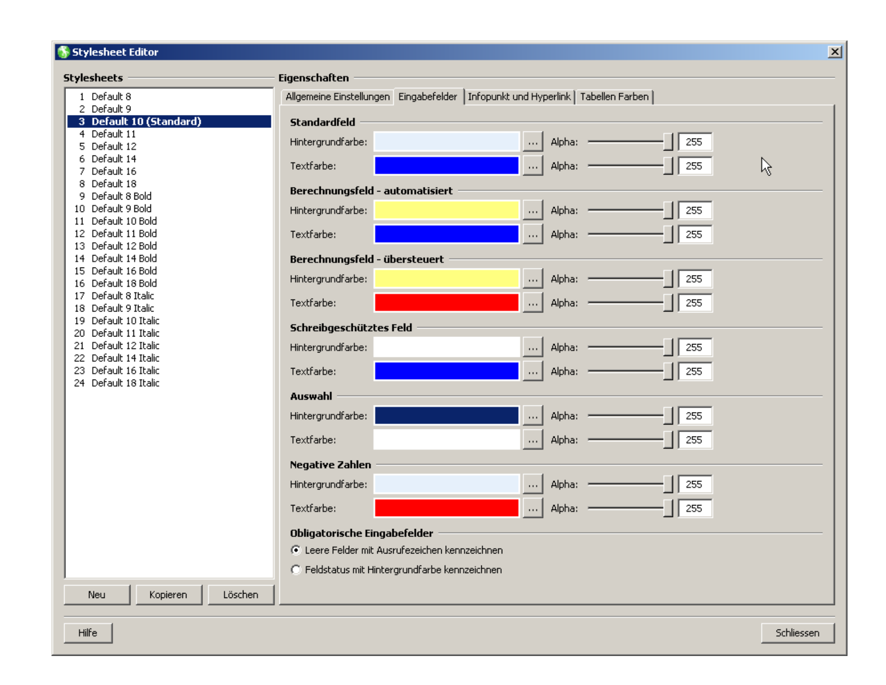
Farbe beim Überfahren mit der Maus ist die Randfarbe, wenn sich die Maus innerhalb der Umrandung des Elements befindet.

Farbe des aktiven Elements ist die Randfarbe, wenn das Element aktiv (für die Eingabe von Informationen) ist. Dies ist beispielsweise dann der Fall, wenn das Element angeklickt worden ist.

Randbreite stellt die Breite des Randes in points ein. Grundwert ist **1**.

Wenn das Kontrollkästchen **Fadingeffekt Rand** eingeschaltet ist, wird beim Überfahren mit der Maus und dem Aktivieren des Elements der Rand graduell ein- und ausgeblendet.

Fig. 4-169 *Der Stilvorlageneditor, Einstellungen für Eingabefelder 111*



In den **Einstellungen für Eingabefelder** wird die Hintergrund- und Textfarbe für verschiedene Zustände des Felds kontrolliert. **Hintergrundfarbe** ist die Farbe des Feldhintergrunds, sofern sie nicht in den Eigenschaften des Elements übersteuert worden ist. **Textfarbe** ist die Farbe des Texts, sofern sie nicht übersteuert worden ist.

Die Einstellung **Standardfeld** ist, wie der Name sagt, die Grundstellung für das (leere) Element.

Berechnungsfeld automatisiert ist die Einstellung für den Fall, wenn im Feld das Resultat einer Berechnung angezeigt wird.

Berechnungsfeld übersteuert ist die Einstellung für den Fall, wenn im Feld das Resultat einer Berechnung angezeigt wird, diese aber durch eine Eingabe übersteuert worden ist.

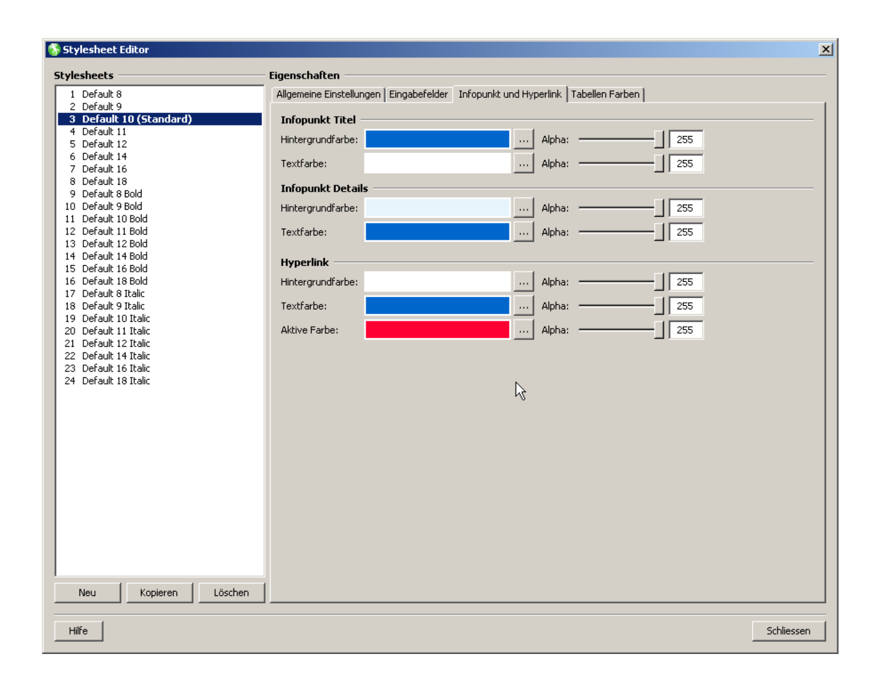
Schreibgeschütztes Feld ist die Einstellung für den Fall, wenn das Feld über die Eigenschaft **schreibgeschützt** nicht beschreibbar gemacht worden ist.

Auswahl ist die Einstellung für den Fall, wenn das Feld ein Auswahlfeld ist.

Negative Zahlen ist die Einstellung für den Fall, dass der Feldwert negativ ist.

Obligatorische Eingabefelder kontrolliert die Anzeige für Felder, welche als obligatorisch gekennzeichnet worden sind. Wenn die Option **Leere Felder mit Ausrufezeichen kennzeichnen** gewählt ist, werden diese Mussfelder mit einem Ausrufezeichen markiert. Wenn die Option **Feldstatus mit Hintergrundfarbe kennzeichnen** gewählt ist, erscheinen drei weitere Farbauswahlen. **Leeres Feld** bestimmt die Farbe des Felds wenn es leer ist. **Ausgefülltes Feld** zeigt die Farbe des ausgefüllten Felds und **Leeres Feld (hervorgehoben)** zeigt die Farbe für ein markiertes leeres Feld.

Fig. 4-170 *Der Stilvorlageneditor, Einstellungen für Infopunkte und Hyperlinks 112*



In den **Einstellungen für Infopunkte und Hyperlinks** wird die Hintergrund- und Textfarbe für verschiedene Zustände von Infopunkten und Hyperlinks kontrolliert. **Hintergrundfarbe** ist die Farbe des Feldhintergrunds, sofern sie nicht in den Eigenschaften des Elements übersteuert worden ist. **Textfarbe** ist die Farbe des Texts, sofern sie nicht übersteuert worden ist.

Infopunkt Titel ist die Einstellung für die Titelleiste von aufgeklappten Inforpunkten.

Infopunkt Details ist die Einstellung für die Anzeigefläche von Infopunkten.

Hyperlink ist die Einstellung für Hyperlinks. Hier kommt neben Hintergrund- und Textfarbe die Einstellung für die Aktive Farbe hinzu, welche beim Aktivieren des Links angewandt wird.

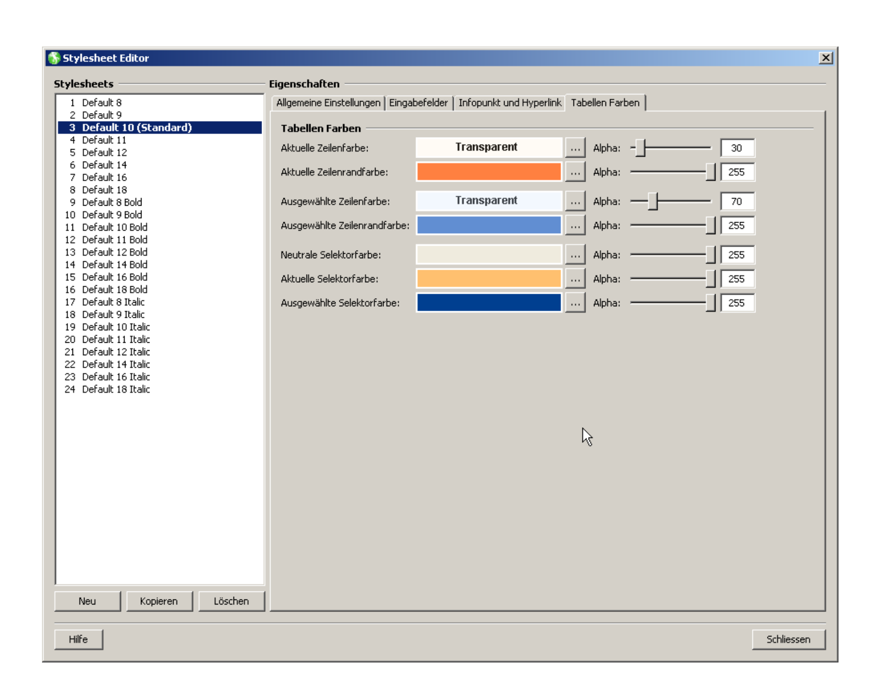


Fig. 4-171 *Der Stilvorlageneditor, Einstellungen für Tabellenfarben 113*

In den **Einstellungen für Tabellen Farben** wird die Farbe für Zeilenhintergrund und Selektor kontrolliert.

Aktuelle Zeilenfarbe ist die Markierung für die aktuelle Zeile (ein Feld dieser Zeile ist aktiv). **Aktuelle Zeilenrandfarbe** ist die Farbe des Rands der aktuellen Zeile.

Ausgewählte Zeilenfarbe ist die Markierung für die ausgewählte Zeile (Zeile via Selektor selektiert). **Ausgewählte Zeilenrandfarbe** ist die Farbe des Rands der ausgewählten Zeile.

Neutrale Selektorfarbe ist die Farbe des Selektors, wenn er nicht gewählt oder aktiv ist. **Aktuelle Selektorfarbe** ist die Markierung für

den aktuellen Selektor der Zeile (ein Feld dieser Zeile ist aktiv). **Ausgewählte Selektorfarbe** ist die Farbe des Rands der ausgewählten Zeile (Zeile via Selektor selektiert).

4.7.2 Stilvorlagen importieren und exportieren

Stilvorlagen werden über die Funktionen **Stylesheet importieren** und **Stylesheet exportieren** importiert und exportiert. Dabei öffnet sich ein Datei Öffnen (beim importieren) bzw. Datei Speichern (beim exportieren) Dialog welcher die Wahl des Dateinamens zulässt.

4.8 Formular- und Sicherheitseinstellungen

4.8.1 Einleitung

Der letzte Schritt bevor das Formular freigegeben und veröffentlicht werden kann, ist die Konfiguration der Formular- und Sicherheitseinstellungen. Diese Einstellungen betreffen verschiedene Aspekte für den einfachen Gebrauch des Formulars (wie die Konfiguration der Aktions-Schaltflächen), aber auch die Voraussetzungen für die Einbettung in Arbeitsabläufe und die notwendigen Vorkehrungen für Archivierung.

Diese Einstellungen werden im **Formulareinstellungen** Fenster vorgenommen. Das Fenster öffnet sich mit dem Menubefehl **Extras —> Formulareinstellungen**.

Die Formulareinstellungen bestehen aus 7 Dialogen, welche über die Auswahlliste am linken Rand des Fensters gewählt werden. Diese Dialoge sind nachstehend erläutert.

Da die Grundeinstellungen für diese Dialoge weitgehend leere Felder umfassen, ist diese Konfigurierung für jedes Formular notwendig.

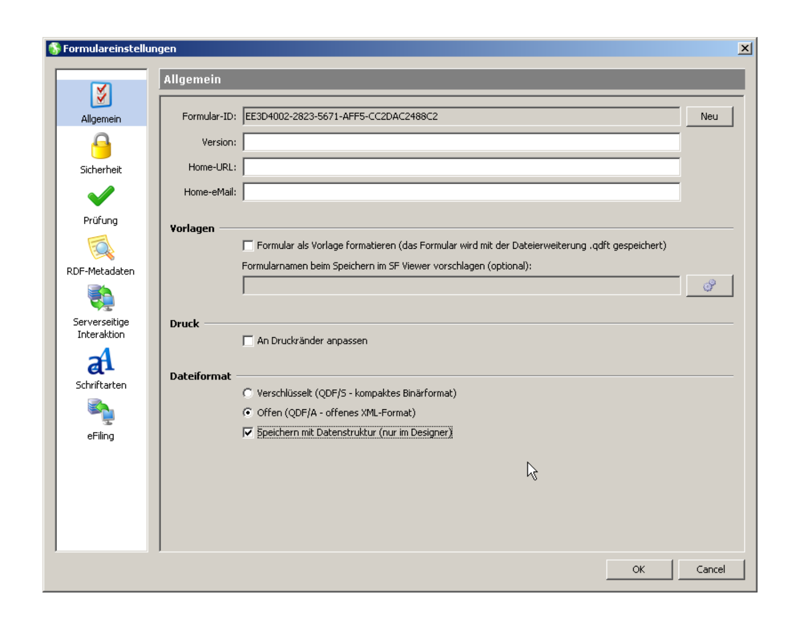
Hinweis: *Die Resultate der Konfigurierung müssen im Snapform Viewer kontrolliert werden, da der Snapform Designer gewisse Funktionen nicht darstellen kann.*

Hinweis: *Einige Einstellungen müssen mit den gegebenen Server-Konfigurationen abgestimmt werden. Hinweise auf Server-seitige Applikationen, deren Funktion und Konfiguration liegen ausserhalb des Bereichs dieses Handbuchs. Für diese Funktionalitäten wird auf die Dokumentation zu den betreffenden Applikationen verwiesen.*

4.8.2 Allgemeine Einstellungen

Allgemeine Einstellungen (siehe Fig. 4-172) ist der Dialog, mit welchem sich das Fenster Formulareinstellungen öffnet. Das Fenster ist in verschiedene Zonen unterteilt: Generelle Angaben, Vorlagen, Druck und Dateiformat.

Fig. 4-172 *Das Fenster Formulareinstellungen: Allgemeine Einstellungen 114*



4.8.2.1 Dokument-Angaben

Die Dokument-Angaben betreffen das eigentliche Dokument, und dienen beispielsweise zu dessen Identifikation in Formular-Managementsystemen.

Formular-ID

Die Formular-ID ist ein eindeutiger Identifier im GUID-Format, welcher das betreffende Dokument weltweit eindeutig identifizieren kann. Dieser Wert wird beim Erstellen des Dokuments erzeugt, und bleibt mit diesem gespeichert.

Mit Anklicken von **Neu** kann eine neue Formular-ID erzeugt werden. Ein Dialog fragt, ob die ID gewechselt werden soll, und bei Bestätigung wird dies ausgeführt.

Version

In diesem Textfeld kann die Version des Dokuments frei eingegeben werden. Dieser Wert erscheint zudem in der Metadaten sowie in der Dateiinformation.

Home-URL

URL des Herausgebers des Formulars. Dies kann zur Aktualisierung des Formulars oder zur Kontaktaufnahme mit dem Herausgeber benutzt werden.

Home-eMail

E-Mail-Adresse des Herausgebers. Diese Adresse kann zur Kontaktaufnahme mit dem Hersteller oder dessen Support benutzt werden.

4.8.2.2 Vorlagen

Das Formular kann als Vorlage gespeichert werden. Damit wird das Formular beim Öffnen immer rückgesetzt, und es kann zudem nicht unter dem gleichen Namen gespeichert werden. Dies ist insbesondere bei zentral in einer Ablage gespeicherten Formularen sinnvoll sein.

Als Vorlage speichern

Mit diesem Kontrollkästchen wird eingestellt, ob das Formular als Vorlage im .qdft-Format gespeichert wird.

Formularnamen beim Speichern vorschlagen

Der Name, unter welchem das (ausgefüllte) Formular im Snapform Viewer gespeichert wird, kann über dieses Feld vorgeschlagen werden. Ein derartiger Vorschlag ist besonders wichtig, wenn das Formular in einem Dokumenten-basierten Arbeitsablauf eingesetzt ist, und der Dateiname eine Bedeutung hat. Der Vorschlag kann entweder statisch (ein fixer Text) oder dynamisch als Resultat einer Formel gestellt werden. Die Struktur des vorgeschlagenen Namens hängt vom jeweiligen Einsatzzweck ab.

Mit Anklicken des Symbols öffnet sich der Formeleditor. Hier kann die Formel zur Berechnung des vorgeschlagenen Dateinamens eingegeben werden. In den Vorschlag einfliessen kann beispielsweise das aktuelle Datum, Name und Vorname, oder ein weiterer eindeutiger

Identifier. Der Formeleditor hat Zugriff auf sämtliche vorhandenen Felder und Funktionen.

4.8.2.3 Druck

An Druckränder anpassen

Dieses Kontrollkästchen bestimmt, ob das Formular beim Ausdruck so skaliert wird, dass es vollständig innerhalb der Druckränder des gewählten Druckers ausgegeben werden kann. Damit wird sicher gestellt, dass das Formular vollständig ausgedruckt wird.

Wenn in der Nachverarbeitung der ausgedruckten Formulare Scanner (mit Texterkennung) eingesetzt werden, muss sehr sorgfältig abgeklärt werden, ob eine Skalierung überhaupt zulässig ist. Sicherheitshalber sollte in diesem Fall das Kontrollkästchen **An Druckränder anpassen** ausgeschaltet werden, und gegebenenfalls ausreichend Rand im Formular selbst geschaffen werden, damit das Formular vollständig ausgegeben werden kann.

Das Kontrollkästchen muss zwingend ausgeschaltet werden, wenn auf vorgedruckte Garnituren ausgedruckt wird.

4.8.2.4 Dateiformat

Das Dateiformat kann zwar schon beim Erstellen eines neuen Dokuments gewählt werden. Es ist jedoch in vielen Fällen zu diesem Zeitpunkt nicht sicher, ob die Daten komprimiert/verschlüsselt oder offen gelegt gespeichert werden sollen. Daher kann das Dateiformat in den Dokumenteinstellungen definitiv gewählt werden.

Verschlüsselt / Offen

Dieses Optionsfeld bestimmt, ob beim Speichern des Formulars die Daten verschlüsselt/komprimiert, oder offen gelegt in XML-Form gespeichert werden.

Die Option **Verschlüsselt** schreibt das Formular im QDF/S-Format, welches sehr kompakt ist zurück. Damit die Daten ausgelesen werden

können, wird eine spezielle Server-Komponente benötigt (weitere Auskünfte hierüber sind von Ringler Informatik AG erhältlich).

Die Option **Offen** schreibt das Formular im QDF/A-Format, welches weniger kompakt ist, jedoch die Daten für jeglichen XML-Parser (oder eine andere Applikation) zugänglich macht.

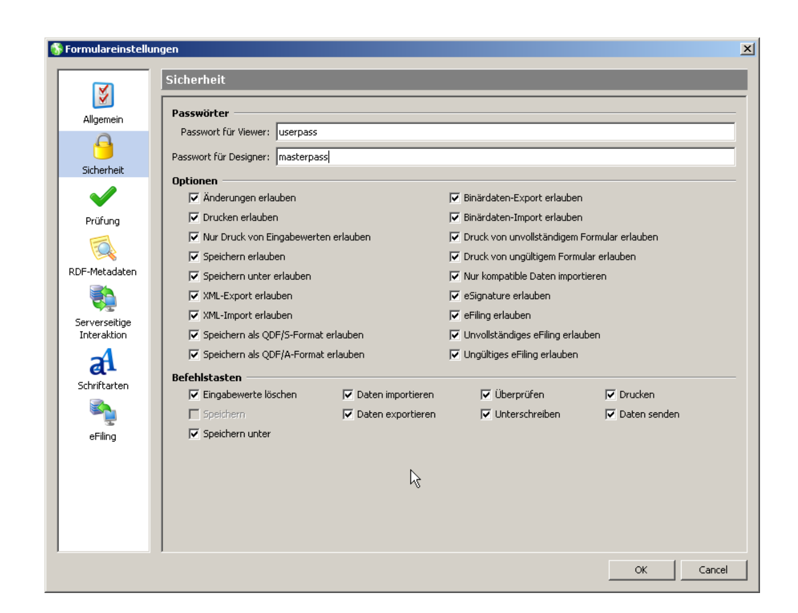
Das Kontrollkästchen **Speichern mit Datenstruktur** ermöglicht die Erzeugung der vollständigen Datenstruktur beim Erstellen und leer Speichern des Formulars. Dies ist notwendig, wenn das leere Formular vorbefüllt werden soll. Ebenso erleichtert diese Option die Arbeit beim Entwickeln der Anbindung des Formulars an Back End-Prozesse.

Hinweis: *Diese Option ist auf Snapform Designer beschränkt. Beim Speichern des Formulars im Snapform Viewer werden nur die ausgefüllten Felder in die Datenstruktur übernommen.*

4.8.3 Sicherheit

Im Dialog **Sicherheit** (siehe Fig. 4-173) sind Dokumenteinstellungen zusammengefasst, welche irgendwie mit der Dokumentensicherheit in Zusammenhang stehen. Dies umfasst den Zugang zum Dokument sowie Benutzungsprivilegien.

Fig. 4-173 *Das Fenster Formulareinstellungen: Sicherheit 115*



Der Dialog Sicherheit ist in drei Zonen unterteil: **Passwörter**, **Optionen** und **Befehlstasten**.

4.8.3.1 Passwörter

Das Dokument kann mit zwei Passwörtern geschützt werden. Die Eingabe in das betreffende Feld erfolgt im Klartext, und die Passwörter werden auch nach dem Speichern und wieder Öffnen im Designer im Klartext angezeigt.

Der Schutz über Passwörter ist bis zu einem gewissen Grad wirksam. Wenn ein weiter gehender Zugangsschutz für das Formular benötigt wird, muss dieser über externe Mechanismen erfolgen. Ringler Informatik AG bietet Unterstützung bei der Implementierung von Schutzmechanismen.

Passwort für User

Dieses Passwort wird beim Öffnen des Formulars durch den regulären Benutzer verlangt. Damit kann der Zugang zum Formular selbst kontrolliert werden. Ohne das User-Passwort kann das Formular im Snapform Viewer nicht geöffnet werden.

Passwort für Designer

Dieses Passwort wird beim Öffnen des Formulars im Designer durch den Entwickler verlangt. Damit kann der Zugang zum Formular und insbesondere dessen Business-Logik kontrolliert werden. Ohne Designer-Passwort kann das Formular im Snapform Designer nicht geöffnet werden.

4.8.3.2 Optionen

In der Gruppe Optionen sind die verschiedenen Berechtigungen für das Formular aufgeführt, und können individuell ein- und ausgeschaltet werden. Diese Optionen werden vom Snapform Viewer berücksichtigt.

Änderungen erlauben

Wenn dieses Kontrollkästchen eingeschaltet ist, kann das Formular in Snapform Designer verändert werden.

Drucken erlauben

Wenn dieses Kontrollkästchen eingeschaltet ist, dann das Formular gedruckt werden.

Nur Druck von Eingabewerten erlauben

Wenn dieses Kontrollkästchen eingeschaltet ist, können nur die Eingabefelder des Formulars gedruckt werden (diese Option ist beispielsweise notwendig, wenn auf vorgedruckte Garnituren gedruckt wird).

Speichern erlauben

Wenn dieses Kontrollkästchen eingeschaltet ist, kann das Formular mit Speichern zurückgeschrieben werden. Es ist jedoch nicht möglich, das Formular mit Speichern unter... unter einem neuen Namen abzustpeichern.

Speichern unter erlauben

Wenn dieses Kontrollkästchen eingeschaltet ist, kann das Formular mit Speichern unter... (auch unter einem anderen Namen) abggespeichert werden.

XML-Export erlauben

Wenn dieses Kontrollkästchen eingeschaltet ist, können die Formulardaten im XML-Format exportiert werden.

XML-Import erlauben

Wenn dieses Kontrollkästchen eingeschaltet ist, können im XML-Format vorliegende Formulardaten importiert werden.

Speichern als QDF/S erlauben

Wenn dieses Kontrollkästchen eingeschaltet ist, kann das Formular im kompakten Binärformat (QDF/S) gespeichert werden.

Speichern als QDF/A erlauben

Wenn dieses Kontrollkästchen eingeschaltet ist, kann das Formular im offen zugänglichen Format (QDF/A) gespeichert werden.

Binärdaten-Export erlauben

Wenn dieses Kontrollkästchen eingeschaltet ist, können die Formulardaten im verschlüsselten Binärdatenformat exportiert werden.

Binärdaten-Import erlauben

Wenn dieses Kontrollkästchen eingeschaltet ist, können im verschlüsselten Binärformat vorliegende Formulardaten importiert werden.

Druck von unvollständigem Formular erlauben

Wenn dieses Kontrollkästchen eingeschaltet ist, kann ein unvollständig ausgefülltes Formular ausgedruckt werden. Die Vollständigkeit wird über eine Prüfregel im Dialog **Prüfung** (siehe Abschnitt 4.8.4.3) festgelegt.

Druck von ungültigem Formular erlauben

Wenn dieses Kontrollkästchen eingeschaltet ist, kann ein ungültig ausgefülltes Formular ausgedruckt werden. Die Gültigkeit wird über eine Prüfregel im Dialog **Prüfung** (siehe Abschnitt 4.8.4.2) festgelegt.

Nur kompatible Daten importieren

Wenn dieses Kontrollkästchen eingeschaltet ist, können Daten nur dann importiert werden, wenn sie mit dem Formular kompatibel sind.

eSignature erlauben

Wenn dieses Kontrollkästchen eingeschaltet ist, kann das Formular mit entsprechend vorhandenen Systemen digital signiert werden.

eFiling erlauben

Wenn dieses Kontrollkästchen eingeschaltet ist, kann das Formular mit dem **eFiling**-Mechanismus (definiert im Dialog eFiling, siehe Abschnitt 4.8.8) elektronisch eingereicht werden.

Die nächsten beiden Optionen werden nur aktiv, wenn dieses Kontrollkästchen eingeschaltet ist.

Unvollständiges eFiling erlauben

Wenn dieses Kontrollkästchen eingeschaltet ist, kann das Formular auch dann elektronisch eingereicht werden, wenn es unvollständig ausgefüllt ist. Die Vollständigkeit wird über eine Prüfregel im Dialog **Prüfung** (siehe Abschnitt 4.8.4.3) festgelegt. Dieses Kontrollkästchen kann nur dann ein- und ausgeschaltet werden, wenn das Kontrollkästchen **eFiling erlauben** eingeschaltet ist.

Ungültiges eFiling erlauben

Wenn dieses Kontrollkästchen eingeschaltet ist, kann das Formular auch dann elektronisch eingereicht werden, wenn es ungültig ausgefüllt ist. Die Gültigkeit wird über eine Prüfregel im Dialog **Prüfung** (siehe Abschnitt 4.8.4.2) festgelegt. Dieses Kontrollkästchen kann nur dann ein- und ausgeschaltet werden, wenn das Kontrollkästchen **eFiling erlauben** eingeschaltet ist.

4.8.3.3 Befehlstasten

Eines der besonderen Merkmale von Snapform ist das Verlagern von Aktions-Schaltflächen vom Formular weg zur Applikation. Die wesentlichen Aktionen (Rücksetzen, Drucken, Speichern, Einreichen) müssen in vielen Formularsystemen für jedes einzelne Formular neu eingebaut und konfiguriert werden. Bei Snapform trägt das Formular nur die Information, dass die betreffende Funktion unterstützt werden soll, und dass der Snapform Viewer diese bereitstellen wird.

Im Bereich Befehlstasten kann gewählt werden, welche Funktionen das Formular unterstützen soll. Damit diese Funktionen auch wirklich ausgelöst werden können, muss das Formular die betreffende Berechtigung enthalten (Einschalten des betreffenden Kontrollkästchens in Bereich **Optionen** (siehe Abschnitt 4.8.3.2)).

Eingabewerte löschen

Wenn dieses Kontrollkästchen eingeschaltet ist, erscheint im Snapform Viewer eine Schaltfläche **Eingabewerte löschen**. Wenn diese Schaltfläche angeklickt wird, wird das Formular rückgesetzt.

Speichern

Wenn dieses Kontrollkästchen eingeschaltet ist, erscheint im Snapform Viewer eine Schaltfläche **Speichern**. Wenn dieses Kontrollkästchen eingeschaltet ist, wird das allenfalls eingeschaltete Kontrollkästchen **Speichern unter** ausgeschaltet. Wenn diese Schaltfläche angeklickt wird, wird das Formular unter dem selben Namen gespeichert.

Speichern unter

Wenn dieses Kontrollkästchen eingeschaltet ist, erscheint im Snapform Viewer eine Schaltfläche **Speichern unter**. Wenn dieses Kontrollkästchen eingeschaltet ist, wird das allenfalls eingeschaltete Kontrollkästchen **Speichern** ausgeschaltet. Wenn diese Schaltfläche angeklickt wird, wird das Formular unter einem anderen (vorgeschlagenen oder vollständig einzugebenden) Namen gespeichert.

Daten importieren

Wenn dieses Kontrollkästchen eingeschaltet ist, erscheint im Snapform Viewer eine Schaltfläche **Daten importieren**. Wenn diese Schaltfläche angeklickt wird, werden Daten in das Formular importiert.

Daten exportieren

Wenn dieses Kontrollkästchen eingeschaltet ist, erscheint im Snapform Viewer eine Schaltfläche **Daten exportieren**. Wenn diese Schaltfläche angeklickt wird, werden die Daten exportiert.

Überprüfen

Wenn dieses Kontrollkästchen eingeschaltet ist, erscheint im Snapform Viewer eine Schaltfläche **Formular überprüfen**. Wenn diese Schaltfläche angeklickt wird, werden die Daten gemäss den definierten Prüfregeln überprüft.

Unterschreiben

Wenn dieses Kontrollkästchen eingeschaltet ist, erscheint im Snapform Viewer eine Schaltfläche **Formular unterschreiben**. Wenn diese

Schaltfläche angeklickt wird, wird der Prozess zum Digital Signieren des Formulars gestartet.

Drucken

Wenn dieses Kontrollkästchen eingeschaltet ist, erscheint im Snapform Viewer eine Schaltfläche **Drucken**. Wenn diese Schaltfläche angeklickt wird, wird das Formular auf dem Standard-Drucker ausgedruckt.

Daten senden

Wenn dieses Kontrollkästchen eingeschaltet ist, erscheint im Snapform Viewer eine Schaltfläche **Daten senden**. Wenn diese Schaltfläche angeklickt wird, werden die Formulardaten an den definierten Server übermittelt (eFiling).

Wenn alle Kontrollkästchen gewählt sind, zeigt sich das Formular im Snapform Viewer wie in Fig. 4-174 dargestellt.

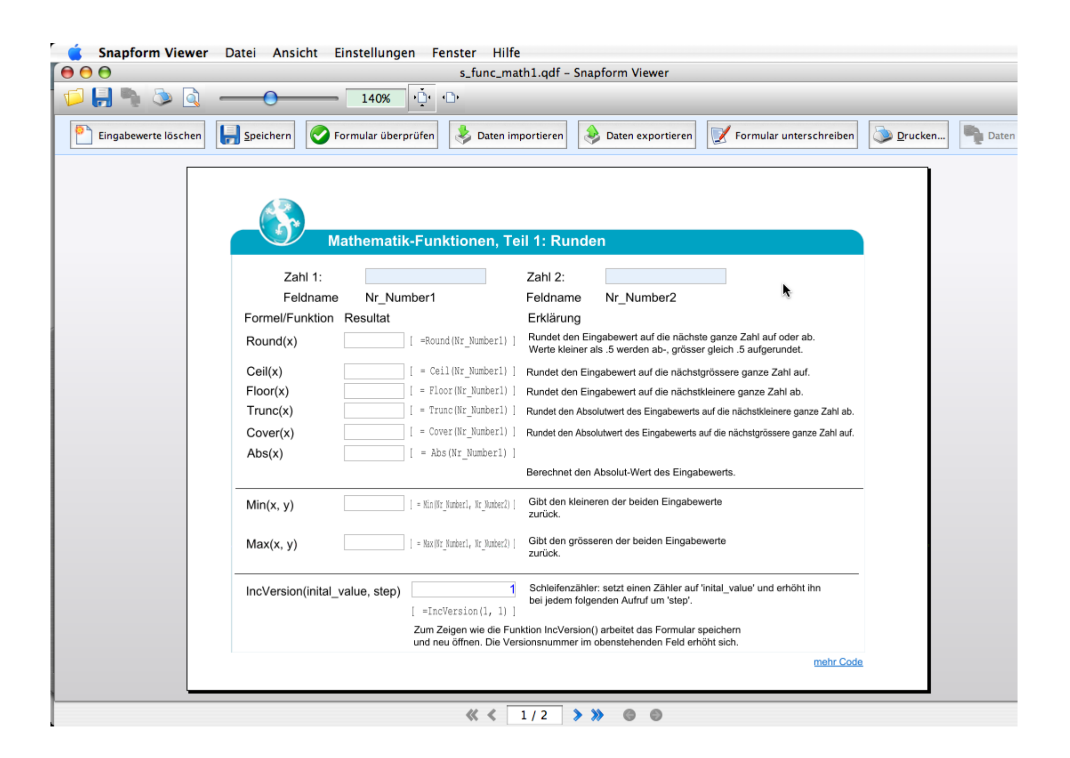


Fig. 4-174 *Die Aktions-Werkzeugleiste in Snapform Viewer*

Unterhalb der normalen Werkzeugleiste erscheint eine neue Werkzeugleiste mit den Aktions-Schaltflächen, welche in den Formulareinstellungen aktiviert worden sind.

4.8.4 Prüfung

Im Dialog **Prüfung** (siehe Fig. 4-175) sind Dokumenteinstellungen zusammengefasst, welche die Gültigkeit des Formulars oder der Daten betreffen.

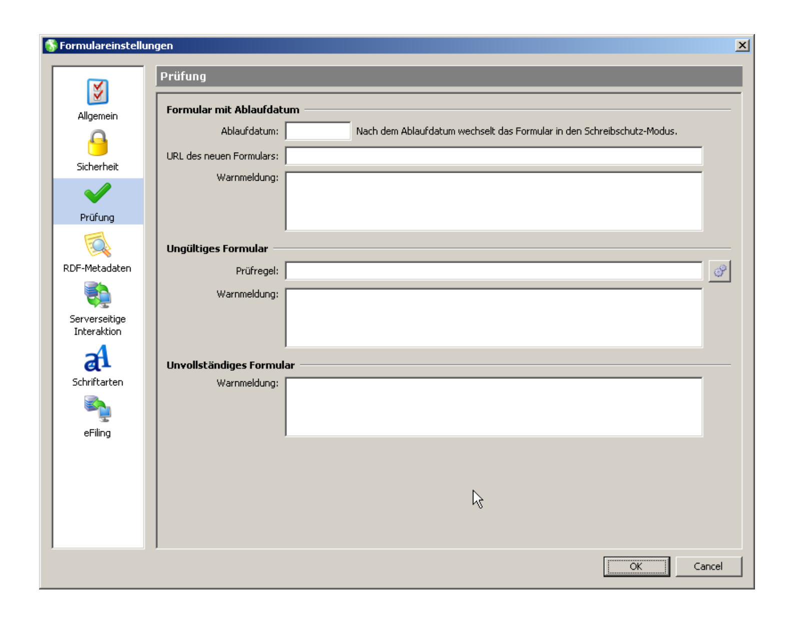


Fig. 4-175 *Das Fenster Formulareinstellungen: Prüfung 116*

Der Dialog **Prüfung** ist in drei Zonen unterteil: **Formular mit Ablaufdatum**, **Ungültiges Formular** und **Unvollständiges Formular**.

4.8.4.1 Formular mit Ablaufdatum

Es ist möglich, das Formular mit einem Ablaufdatum zu versehen. dies ist insbesondere dann sinnvoll, wenn die darin enthaltenen Daten für eine bestimmte Zeitspanne relevant sind, und für spätere Daten ein anderes Formular benutzt werden muss.

Ablaufdatum

In diesem Feld wird das Ablaufdatum eingegeben. Das Datum wird mit der Systemuhr des Rechners verglichen, und wenn das aktuelle Datum nach dem Ablaufdatum ist, wird das Formular schreibgeschützt geöffnet, und es können keine Eingaben mehr vorgenommen werden.

Sollte eine "härtere" Kontrolle des Ablaufdatums als durch die Systemuhr bestimmt, muss eine externe Lösung eingesetzt werden. Ringler Informatik AG bietet Unterstützung bei der Implementation derartiger Lösungen.

URL des neuen Formulars

Wenn ein Ablaufdatum gesetzt wird, sollte auch ein Verweis auf das aktuell gültige Formular vorhanden sein. Dieser Verweis kann ein URL sein, welcher in diesem Feld einzugeben ist.

Warnmeldung

Wenn ein abgelaufenes Formular geöffnet wird, muss der Benutzer auch darauf hingewiesen werden. Der entsprechende Text ist anwendungsabhängig, und kann daher nicht vom System festgelegt werden. In diesem Feld kann der Text eingegeben werden, welcher gegebenenfalls in einer Systemmeldung erscheint.

4.8.4.2 Ungültige Formulare

Um insbesondere bei elektronischem Einreichen der Formulare zu verhindern, dass ungültige Daten übermittelt werden (welche unter Umständen das Verarbeitungssystem lahmlegen könnten), kann das Formular auf Gültigkeit geprüft werden. Diese Prüfung erfolgt entsprechend einer Prüfregel, welche in dieser Zone festgelegt wird.

Prüfregel

Die Prüfregel ist eine Formel, welche entweder **TRUE** oder **FALSE** zurückgeben muss. Ist der Wert **TRUE**, ist das Formular gültig, anderenfalls ist es ungültig.

Mit Anklicken des Formel-Symbols öffnet sich der Formeleditor zur Eingabe der Formel. Es stehen sämtliche Feldobjekte und Funktionen zur Verfügung.

Warnmeldung

Wenn das Formular als ungültig bewertet wird, muss der Benutzer benachrichtigt werden. Dies geschieht über die frei aufbaubare Warnmeldung in diesem Feld. Der Text wird gegebenenfalls in einer Systemmeldung wiedergegeben.

4.8.4.3 Unvollständige Formulare

Im Gegensatz zur Gültigkeit des Formulars ist die Vollständigkeit einfach zu prüfen, indem festgestellt wird, ob sämtliche Felder bei denen die Feldeigenschaft erforderlich eingeschaltet ist, einen Wert haben. Das bedeutet, dass keine besondere Prüfregel vorhanden sein muss.

Warnmeldung

Wenn das Formular als unvollständig bewertet wird, muss der Benutzer benachrichtigt werden. Dies geschieht über die frei aufbaubare Warnmeldung in diesem Feld. Der Text wird gegebenenfalls in einer Systemmeldung wiedergegeben.

4.8.5 RDF-Metadaten

Für die Verwaltung und Archivierung von Dokumenten sind korrekt aufgebaute Metadaten eine fundamentale Voraussetzung. Ein programmgesteuertes Aufbauen stellt dabei sicher, dass zusätzliche Fehlerquellen durch nachträgliches Erfassen ausgeschaltet werden.

In Snapform können Elemente der Dublin Core Metadata Initiative Definition 1.1 entweder statisch oder über Formeln berechnet erzeugt werden.

Im Dialog **RDF-Metadaten** (siehe Fig. 4-176) sind die RDF-Metadaten-Elemente, welche für die Formulare relevant sind, zusammengestellt.

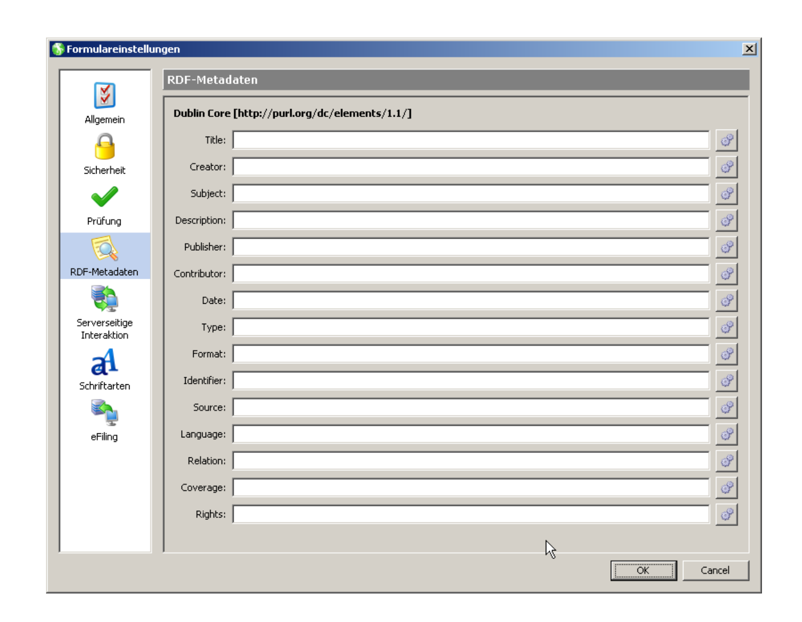


Fig. 4-176 *Das Fenster Formulareinstellungen: RDF-Metadaten 117*

Der Dialog **RDF-Metadaten** umfasst Eingabefelder für die verschiedenen Metadaten-Elemente. Für eine genauere Definition der jeweiligen Elemente wird auf die Dokumentation der Dublin Core Metadata Initiative (http://www.dublincore.org) verwiesen.

Die Werte können entweder direkt als statischer Text, oder als Resultat einer Formel angegeben werden. Bei jedem Element wird durch Anklicken des Formel-Symbols der Formeleditor zur Eingabe der Formel geöffnet. Es stehen sämtliche Feldobjekte und Funktionen zur Verfügung.

4.8.6 Serverseitige Interaktion

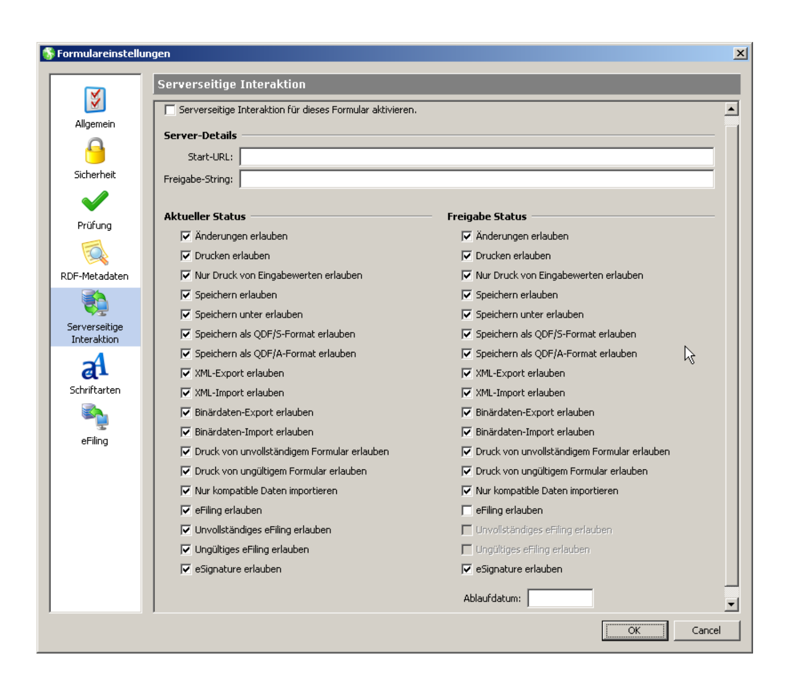
Snapform bietet die Möglichkeit, über eine sogenannte Freigabe, die Formular-Berechtigungen (siehe Abschnitt 4.8.3.2) über einen Serverbefehl zu ändern. Denkbare Anwendungen sind beispielsweise das Schützen des Formulars nach dem elektronischen Einreichen, oder die Freigabe von zusätzlichen Funktionen nach Freigabe durch ein Bezahlsystem.

Die serverseitige Interaktion erfolgt derart, dass ein bestimmter URL angerufen wird, und dieser einen String zurückschickt. Wenn dieser dem definierten Freigabestring entspricht, erfolgt die Freigabe, anderenfalls passiert nichts (es sei denn, das Ereignis würde in einer Formel ausgewertet und eine Nachricht angezeigt).

Hinweis: *Der Begriff "Freigabe" bedeutet in diesem Zusammenhang ein Verändern des Status des Formulars, welcher durch eine Interaktion mit dem Server ausgelöst wird. Es bedeutet jedoch nicht, dass das Formular anschliessend mehr Rechte hat.*

Im Dialog **Serverseitige Interaktion** (siehe Fig. 4-177) sind die für die Serverkommunikation wichtigen Angaben zusammengefasst.

Fig. 4-177 *Das Fenster Formulareinstellungen: Serverseitige Interaktion 118*



Der Dialog **Serverseitige Interaktion** umfasst die Angaben zur Freigabe sowie den Einstellungs-Status für das Formular vor und nach der Freigabe.

Serverseitige Interaktion für dieses Formular aktivieren

Dieses Kontrollkästchen ist der "Hauptschalter" für die serverseitige Interaktion. Wenn es eingeschaltet ist, ist der Mechanismus aktiv, ansonsten werden alle entsprechenden Eintstellungen ignoriert.

In der Zone **Server-Details** befinden sich die Angaben für die Server-Anfrage.

Start-URL

Dies ist der URL, unter welchem die Server-Anfrage für die Freigabe gestartet wird.

Freigabestring

Dies ist der String, welcher für eine Freigabe durch den Server zurückgeschickt werden muss. Wenn dieser String empfangen wird, erfolgt die Freigabe.

In der Zone **Aktueller Status** sind die bereits in den Sicherheitseinstellungen festgelegten Optionen (siehe Abschnitt 4.8.3.2) wiederholt. Sie können auch in diesem Dialog verändert werden.

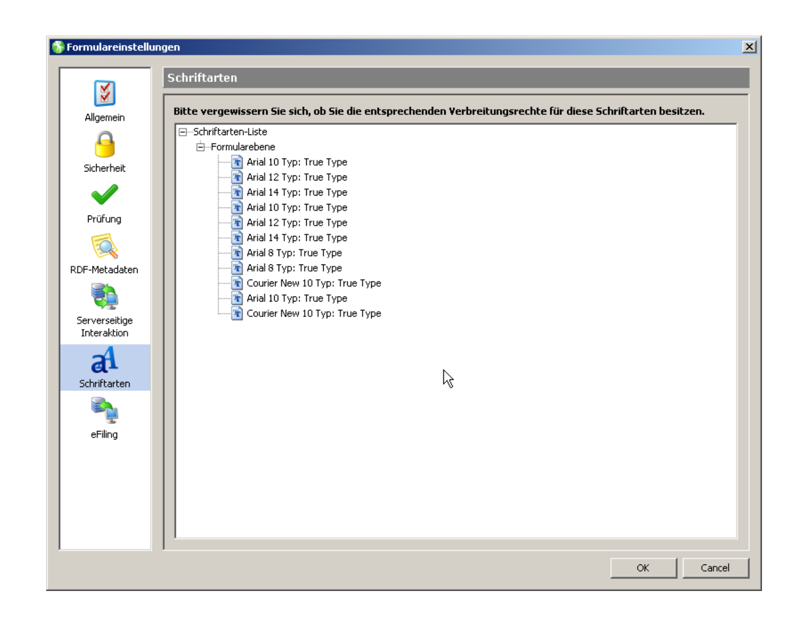
In der Zone **Freigabe Status** sind die Optionen für das Formular nach der Freigabe angegeben. Diese Optionen müssen in diesem Dialog eingestellt werden.

Neben den Optionen ist ein Feld **Ablaufdatum** vorhanden, in welchem ein Ablaufdatum für den fregegebenen Zustand gesetzt werden kann (siehe auch Abschnitt 4.8.4.1),

4.8.7 Schriftarten

Im Dialog **Schriftarten** (siehe Fig. 4-178) sind die vom Formular benutzten (bzw. über Stilvorlagen definierten) Schriften aufgeführt.

Fig. 4-178 *Das Fenster Formulareinstellungen: Schriftarten 119*



Der Dialog **Serverseitige Interaktion** ist eine Zusammenfassung der benutzten Schriften, und dient primär dem Entwickler zur Hilfe bei der Abklärung, ob Lizenzbedingungen für die verwendeten Schriften eingehalten werden können.

Hinweis: *Es liegt in der Verantwortung des Formularentwicklers, sicher zu stellen, dass die mit den verwendeten Schriften verbundenen Lizenzbedingungen eingehalten werden. Ringler Informatik AG lehnt jegliche Verantwortung für Lizenzverletzungen ab.*

Dieser Dialog dient ausschliesslich zur Anzeige von Informationen.

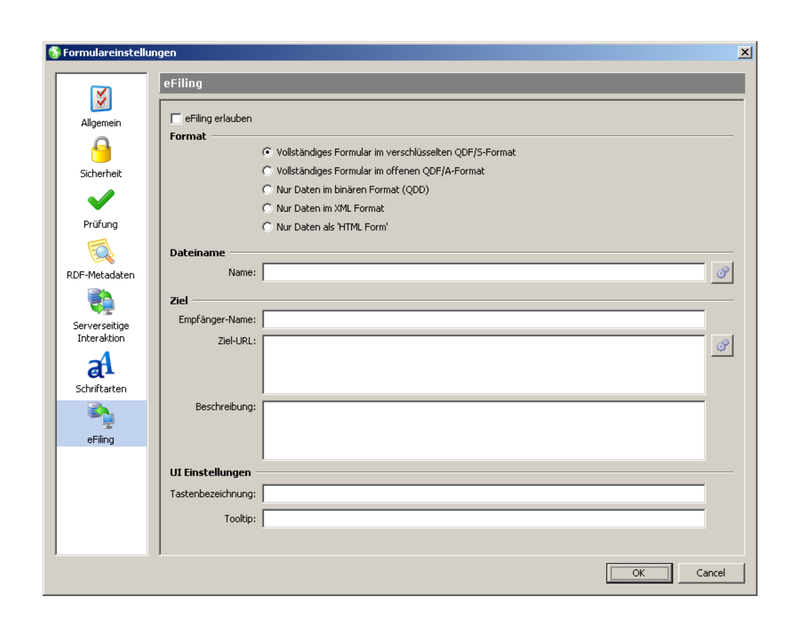
4.8.8 eFiling (Datenübertragung)

Einer der grossen Vorzüge von elektronischen Formularen ist die Möglichkeit, medienbruchfrei die Informationen in den betreffenden Geschäftsprozess zu übertragen. Diese Datenübertragung wird besonders im Behördenumfeld als "eFiling" bezeichnet.

Es ist klar, dass für ein erfolgreiches eFiling das Formular und ein entsprechender serverseitiger Empfangsprozess aufeinander abgestimmt sein müssen. Das bedeutet auch für den Formularentwickler, dass er mit den jeweiligen IT-Abteilungen zusammenarbeiten muss.

Im Dialog **eFiling** (siehe Fig. 4-179) sind die für Datenübertragung zum elektronischen Einreichen wichtigen Informationen zusammengesfasst.

Fig. 4-179 *Das Fenster Formulareinstellungen: eFiling 120*



Der Dialog **eFiling** ist in die Zonen **Format**, **Dateiname**, **Ziel** und **UI Einstellungen** aufgeteilt.

eFiling erlauben

Dieses Kontrollkästchen ist der “Hauptschalter” für as elektronische Einreichen des Formulars. Wenn es eingeschaltet ist, ist der Mechanismus aktiv, ansonsten werden alle entsprechenden Eintstellungen ignoriert.

4.8.8.1 Format

Das eFiling kann über verschiedene Datenformate erfolgen. Das bevorzugte Format hängt einerseits von rechtlich/organisatorischen Randbedingungen, andererseits von der Serverarchitektur des Back End-Systems ab.

Die Formatauswahl erfolgt über einen Satz von Optionsfeldern:

Vollständiges Formular im verschlüsselten QDF/S-Format

Das Formular wird im verschlüsselten QDF/S-Format gespeichert und als Ganzes übertragen. Damit diese Übermittlungsart möglich ist, müssen die Formular-Optionen **eFiling erlauben** und **Speichern im QDF/S-Format erlauben** eingeschaltet sein. Diese Übertragungsart verlangt zur Weiterverarbeitung besondere Werkzeuge zum Auslesen der Daten.

Vollständiges Formular im offenen QDF/A-Format

Das Formular wird im offenen QDF/A-Format gespeichert und als Ganzes übertragen. Damit diese Übermittlungsart möglich ist, müssen die Formular-Optionen **eFiling erlauben** und **Speichern im QDF/A-Format erlauben** eingeschaltet sein. Diese Übertragungsart erlaubt das Auslesen der Daten im Klartext.

Nur Daten im Binären Format (QDD)

Die Formulardaten werden im binären QDD-Format übertragen, ohne dass das Formular mitgeschickt wird. Damit diese Übermittlungsart möglich ist, müssen die Formular-Optionen **eFiling erlauben** und **Binärdaten-Export erlauben** eingeschaltet sein. Diese Übertragungsart verlangt zur Weiterverarbeitung besondere Werkzeuge zum Auswerten der Daten.

Nur Daten im XML-Format

Die Formulardaten werden im XML-Format übertragen, ohne dass das Formular mitgeschickt wird. Damit diese Übermittlungsart möglich ist, müssen die Formular-Optionen **eFiling erlauben** und **XML-Export erlauben** eingeschaltet sein. Diese Übertragungsart benötigt zur Weiterverarbeitung keine besondere Werkzeuge zum Auswerten der Daten.

Nur Daten als HTML-Form

Die Formulardaten werden im als HTML POST Method übertragen, ohne dass das Formular mitgeschickt wird. Damit diese Übermittlungsart möglich ist, muss die Formular-Optionen **eFiling erlauben** eingeschaltet sein. Mit dieser Übertragungsart können auf Empfangsseite dieselben Skripts und Applikaitonen benutzt werden, wie sie mit HTML-Formularen eingesetzt werden.

4.8.8.2 Dateiname

Wenn Dateien übertragen werden, muss der Dateiname vorher festgelegt werden. Dies kann entweder ein statischer Wert sein (nur beschränkt empfohlen) oder ein dynamisch durch eine Formel aufgebauter Name sein. Mit Anklicken des Formel-Symbols öffnet sich der Formeleditor zur Eingabe der Formel. Es stehen sämtliche Feldobjekte und Funktionen zur Verfügung.

4.8.8.3 Ziel

In dieser Zone wird die Information für das Empfangssystem eingegeben.

Empfänger-Name

In diesem Feld kann eine zusätzliche Referenz für die Weiterverarbeitung (z.B. der Name der Abteilung) eingegeben werden.

Ziel-URL

Dies ist der URL, an welchen die Übertragung gehen soll. Dies kann entweder ein statischer Wert oder ein dynamisch durch eine Formel bestimmter Wert sein. Mit Anklicken des Formel-Symbols öffnet sich der Formeleditor zur Eingabe der Formel. Es stehen sämtliche Feldobjekte und Funktionen zur Verfügung.

Beschreibung

Dies ist ein beschreibender Text für die Datenübertragung.

4.8.8.4 UI Einstellungen

In dieser Zone kann die im Snapform Viewer gegebenenfalls dargestellte und über die Sicherheitseinstellungen kontrollierte Schaltfläche **eFiling** (siehe Abschnitt 4.8.3.3) modifiziert werden, da der Prozess des eFilings je nach Arbeitsablauf eine unterschiedliche Bedeutung hat (z.B. "Bezahlen", "Einreichen", "Übermitteln").

Tasten Bezeichnung

In diesem Feld wird die modifizierte Bezeichnung der Snapform Viewer-Schaltfläche **eFiling** eingegeben. Keine Eingabe bedeutet, dass die ursprüngliche Bezeichnung **eFiling** benutzt wird.

Tooltip

In diesem Feld wird der modifizierte Tooltip-Text eingegeben, welcher im Snapform Viewer der Schaltfläche **eFiling** zugewiesen wird. Keine Eingabe bedeutet, dass der Originaltext verwendet wird.

4.9 Freigabe und Veröffentlichung

Das Formular ist mit den Schritten in Abschnitt 4.8 für die Veröffentlichung vorbereitet worden. Damit es nun mit gutem Gewissen veröffentlicht werden kann, muss es eingehend getestet werden. Diese Tests müssen auch eine allfällige Datenübertragung und Datenbankanbindung umfassen.

Es empfiehlt sich, in jedem Fall ab einer gewissen Anzahl zu veröffentlichenden Formularen, einen formalisierten Freigabeprozess aufzubauen. Dies ist insbesondere notwendig, wenn die Formulare in einem Umfeld eingesetzt werden, in welchem erhöhte Anforderungen an die Dokumentation von Prozessen gestellt werden.

Der Freigabeprozess ist im wesentlichen eine organisatorische Angelegenheit, und kann daher durch den Snapform Designer direkt nicht weiter unterstützt werden. Werkzeuge des Formular- und Dokumentenmanagements können hingegen die Abläufe kontrollieren und dokumentieren. Für die Auswahl und Implementierung derartiger Prozesse steht Ringler Informatik AG gerne zur Verfügung.

Die eigentliche Veröffentlichung ist auf technischer Ebene sehr einfach. Die Formulare können wie irgendwelche andere Dokumente in einer Web-Umgebung für das Intranet oder Internet bereit gestellt werden. Wenn Zugangskontrollen notwendig sind, sind diese wie für jeglichen anderen Zugriff auf Webinhalte einzurichten. Snapform-Formulare sind immer ein File-Download, was bedeutet, dass auch serverseitig keine weiteren Anpassungen und Konfigurationen notwendig sind (dies gilt natürlich nur für die Bereitstellung von Formularen; für die Entgegennahme von ausgefüllten Formularen und Daten sind wohl Anpassungen notwendig).

Wenn die Verteilung des Formulars auf Datenträger erfolgen soll, ist das Vorgehen ebenfalls unkompliziert. Snapform-Formulare sind eigenständige Dateien, und können wie irgendwelche anderen Dateien auf Datenträger oder Fileserver kopiert werden.

Die Verteilung von eigenen Snapform-Formularen ist grundsätzlich frei möglich, und die Rechte zu dieser Verteilung sind aus Siche von Snapform durch die Lizenz von Snapform Designer abgedeckt. Die Weitergabe von Snapform Viewer (zum Beispiel auf Datenträger) ist zulässig, aber es muss dafür gesorgt werden, dass der Benutzter aufgefordert wird, jeweils die neueste Version zu beschaffen.

4.10 Formularmanagement

Das Formularmanagement ist in Bereichen, welche höheren Ansprüchen an Sicherheit oder Nachvollziehbarkeit genügen müssen sehr wichtig. Ein sauberes Versionenmanagement ist unumgänglich. Das Versionenmanagement wird in Snapform über die Formular-ID in den Allgemeinen Formulareinstellungen (siehe Abschnitt 4.8.2) sowie über die Metadaten unterstützt (siehe Abschnitt 4.8.5).

Es wird dringend empfohlen, ein den Anforderungen gerechtes Formularmanagement aufzubauen. Für die Definition und Implementation eines derartigen Systems steht Ringler Informatik AG zur Verfügung.

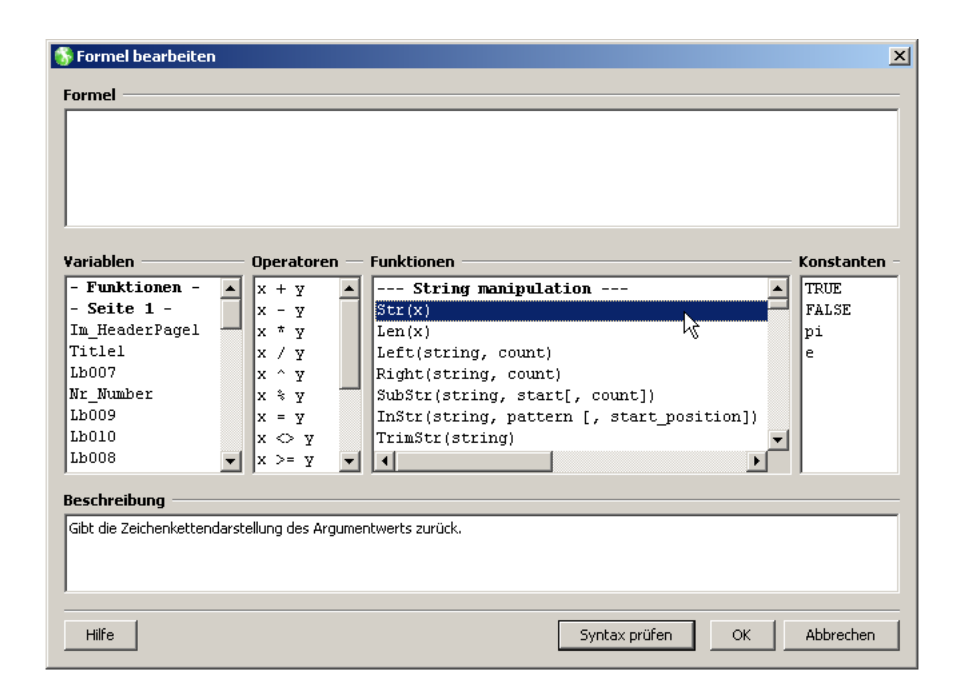
Auf der Eingangsseite ist ein geeignetes Dokumenten- und Archivmanagement notwendig. Die technischen und organisatorischen Anforderungen richten sich nach den internen und externen (gesetzlichen) Vorgaben. Die Formulare können wiederum über eindeutige IDs und Metadaten beitragen.

Snapform-Formulare können dank ihrer XML-Struktur relative einfach in die meisten Dokumentenverwaltungs- und Archivsysteme eingebunden werden. Auch hier bietet Ringler Informatik AG die notwendige Unterstützung.

5 Formelsprache-Referenz

In diesem Kapitel werden die Elemente der Formelsprache detailliert beschrieben. Ausgangspunkt ist der Formeleditor (siehe Fig. 5-1), bei welchem die drei Zonen **Funktionen**, **Operatoren** und **Konstanten** betrachtet werden.

Fig. 5-1 *Der Formeleditor*



Die Benutzeroberfläche des Formeleditors ist in Abschnitt 3.2.8 beschrieben. Hinweise zum Vorgehen beim Einbau von Formeln in das Formular befinden sich in Abschnitt 4.5. Diese Hinweise gelten sinngemäss auch für den Funktionseditor, dessen Benutzeroberfläche identisch ist.

Konstanten sind Werte, welche in Formeln benutzt werden, und immer systemweit den gleichen Wert aufweisen. Die Zahl Pi ist ein gutes Beispiel für eine Konstante.

Operatoren werden auf einem oder zwei Elementen angewandt. Das Resultat wird zurückgegeben und dem Ziel zugewiesen. Ein Beispiel für einen Operator ist die Addition (Symbol **+**),

Funktionen werden auf den Argumenten angewandt. Das Resultat wird zurückgegeben und dem Ziel zugewiesen. Funktionsresultate können zudem mit Operatoren verknüpft werden oder als Argument für eine andere Funktion dienen. Sie sind zudem ein gleichwertiger Ersatz für Beschriftungsfelder, welche einzig als Träger von Formeln gedient haben.

5.1 Konstanten

5.1.1 TRUE

Typ

Boolean

Beschreibung

Dies ist eine Boole'sche Konstante und hat die Bedeutung von "wahr" (entsprechend "zutreffend"). Sie erscheint als Resultat einer logischen Verknüpfung oder als Rückgabewert einer Funktion.

Hinweis: *Diese Konstante kann auch klein geschrieben werden (***true***)*

5.1.2 FALSE

Typ

Boolean

Beschreibung

Dies ist eine Boole'sche Konstante und hat die Bedeutung von "falsch" (entsprechend "nicht zutreffend"). Sie erscheint als Resultat einer logischen Verknüpfung oder als Rückgabewert einer Funktion.

Hinweis: *Diese Konstante kann auch klein geschrieben werden (***false***)*

5.1.3 pi

Typ

Zahl

Beschreibung

Dies ist die Zahl Pi (π), welche dem Verhältnis zwischen Kreisumfang und Kreisdurchmesser entspricht. Der Wert lautet 3,141592653589793 in der in Snapform definierten Genauigkeit.

5.1.4 e

Typ

Zahl

Beschreibung

Dies ist die Zahl e, die Basis des natürlichen Logarithmus. Der Wert lautet 2.718281828459045 in der in Snapform definierten Genauigkeit.

5.1.5 null

Typ

Symbol

Beschreibung

Dies ist ein Symbol für einen nicht existierenden bzw. unzulässigen Wert (z.B. das Resultat einer Division durch 0). Diese Konstante wird Snapform-intern benutzt, ist jedoch für den Formeleditor nicht zugänglich.

5.2 Operatoren

5.2.1 x + y

Typ

numerisch, String (je nach Operandentyp)

Operandentyp

x und y numerisch oder String.

Beschreibung

Addition. Dieser Operator addiert die beiden Operanden und gibt das Resultat zurück.

Das Resultat hängt vom Typ der Operanden ab. Wenn die Operanden numerisch sind (aus einem Zahlenfeld stammen), werden sie mathematisch zusammengezählt. Wenn sie Strings sind (aus einem Textfeld stammen), werden sie aneinandergefügt.

Hinweis: *Vorsicht ist geboten, wenn ein Operand eigentlich wie eine Zahl aussieht, aber nach wie vor ein String ist. In diesem Fall werden die Zahlen aneinandergefügt, und nicht addiert.*

5.2.2 x - y

Typ

numerisch

Operandentyp

x und y numerisch

Beschreibung

Subtraktion. Dieser Operator subtrahiert den Operanden y vom Operanden x und gibt das Resultat zurück. Das Resultat ist, sofern die Operanden numerisch sind, auch numerisch, anderenfalls ist das Resultat leer.

5.2.3 x * y

Typ numerisch

Operandentyp x und y numerisch

Beschreibung Multiplikation. Dieser Operator multipliziert die beiden Operanden und gibt das Resultat zurück. Das Resultat ist, sofern die Operanden numerisch sind, auch numerisch, anderenfalls ist das Resultat leer.

5.2.4 x / y

Typ numerisch

Operandentyp x und y numerisch

Beschreibung Division. Dieser Operator dividiert den Operanden x durch den Operanden y und gibt das Resultat zurück. Das Resultat ist, sofern die Operanden numerisch sind, auch numerisch, anderenfalls ist das Resultat leer.

Hinweis: *Bei einer Division durch 0 (bzw. einem leeren Wert) ist das Resultat leer, und es erscheint keine weitere Warnung, dass die Folgeresultate nicht zulässig sind.*

5.2.5 x ^ y

Typ numerisch

Operandentyp x und y numerisch

Beschreibung

Potenz. Dieser Operator setzt den Operanden x in die Potenz y und gibt das Resultat zurück. Das Resultat ist, sofern die Operanden numerisch sind, auch numerisch, anderenfalls ist das Resultat leer.

5.2.6 x % y

Typ

numerisch

Operandentyp

x und y numerisch

Beschreibung

Restwert. Dieser Operator dividiert den Operanden x durch den Operanden y, bestimmt den Rest (Modulo) und gibt als Resultat zurück. Das Resultat ist, sofern die Operanden numerisch sind, auch numerisch, anderenfalls ist das Resultat leer.

Hinweis: *Bei einer Division durch 0 (bzw. einem leeren Wert) lautet das Resultat "NaN" (Not a Number), und es erscheint keine weitere Warnung, dass die Folgeresultate nicht zulässig sind.*

5.2.7 x = y

Typ

Vergleich

Operandentyp

x und y boolean, numerisch, String, Datum; jeweils vom gleichen Typ

Beschreibung

Vergleicht die beiden Operanden und gibt das Resultat zurück. Wenn die Operanden gleich sind lautet das Resultat **TRUE**, wenn sie verschieden sind, lautet es **FALSE**, und wenn sie von unterschiedlichem Typ sind, wird ein leeres Resultat ausgegeben.

5.2.8 x <> y

Typ

Vergleich

Operandentyp x und y boolean, numerisch, String, Datum; jeweils vom gleichen Typ

Beschreibung Vergleicht die beiden Operanden und gibt das Resultat zurück. Wenn die Operanden verschieden sind lautet das Resultat **TRUE**, wenn sie gleich sind, lautet es **FALSE**, und wenn sie von unterschiedlichem Typ sind, wird ein leeres Resultat ausgegeben.

5.2.9 x >= y

Typ Vergleich

Operandentyp x und y numerisch

Beschreibung Vergleicht die beiden Operanden und gibt das Resultat zurück. Wenn der Operand x grösser oder gleich wie der Operand y ist, lautet das Resultat **TRUE**, wenn er kleiner ist, lautet es **FALSE**.

Hinweis: *Wenn Operanden vom Typ String verglichen werden, ist das Resultat immer* **FALSE**.

5.2.10 x > y

Typ Vergleich

Operandentyp x und y numerisch

Beschreibung Vergleicht die beiden Operanden und gibt das Resultat zurück. Wenn der Operand x grösser als der Operand y ist, lautet das Resultat **TRUE**, wenn er kleiner oder gleich ist, lautet es **FALSE**.

Hinweis: *Wenn Operanden vom Typ String verglichen werden, ist das Resultat immer* **FALSE**.

5.2.11 x <= y

Typ

Vergleich

Operandentyp

x und y numerisch

Beschreibung

Vergleicht die beiden Operanden und gibt das Resultat zurück. Wenn der Operand x kleiner oder gleich wie der Operand y ist, lautet das Resultat **TRUE**, wenn er grösser ist, lautet es **FALSE**.

Hinweis: *Wenn Operanden vom Typ String verglichen werden, ist das Resultat immer* **FALSE**.

5.2.12 x < y

Typ

Vergleich

Operandentyp

x und y numerisch

Beschreibung

Vergleicht die beiden Operanden und gibt das Resultat zurück. Wenn der Operand x kleiner als der Operand y ist, lautet das Resultat **TRUE**, wenn er grösser oder gleich ist, lautet es **FALSE**.

Hinweis: *Wenn Operanden vom Typ String verglichen werden, ist das Resultat immer* **FALSE**.

5.2.13 x AND y

Typ

Logische Verknüpfung

Operandentyp

x und y boolean

Beschreibung

Logisches UND: Vergleicht die beiden Operanden und gibt das Resultat zurück. Wenn beide Operanden **TRUE** sind, lautet das Resultat **TRUE**, wenn einer oder beide **FALSE** sind, lautet es **FALSE**.

Die Operanden sind in der Praxis Vergleichsoperationen

5.2.14 x OR y

Typ

Logische Verknüpfung

Operandentyp

x und y boolean

Beschreibung

Logisches ODER: Vergleicht die beiden Operanden und gibt das Resultat zurück. Wenn einer oder beide Operanden **TRUE** sind, lautet das Resultat **TRUE**, wenn beide **FALSE** sind, lautet es **FALSE**.

Die Operanden sind in der Praxis Vergleichsoperationen

5.2.15 NOT x

Typ

Logische Verknüpfung

Operandentyp

boolean

Beschreibung

Logische Umkehrung: kehrt den Wert des Operanden um und gibt das Resultat zurück. Wenn wenn der Operand **TRUE** ist, lautet das Resultat **FALSE**, wenn er **FALSE** ist, lautet es **TRUE**.

Dieser Operator ist selbstbezogen.

5.3 Funktionen

5.3.1 Abs(x)

Kategorie
Mathematik-Funktion

Kurzbeschreibung
Berechnet den Absolut-Wert des Eingabewerts.

Argumente
x
numerisch, beliebiger Wert

Resultat
numerisch

Erklärung
Gibt den Absolutwert des Eingabewerts zurück. Wenn dieser grösser als **0** ist, sind Eingabe- und Ausgabewert identisch. Im anderen Fall ist der Ausgabewert der Eingabewert multipliziert mit **-1** (was dem Eingabewert ohne negativem Vorzeichen entspricht. Wenn das Argument nicht eine Zahl ist, ist der Ausgabewert der Funktion **null**.

Siehe auch
Andere Mathematik-Funktionen

Beispiel
Siehe func_math1.qdf

5.3.2 Acos(x)

Kategorie
Mathematik-Funktion

Kurzbeschreibung
Gibt den Arcus Cosinus des Eingabewerts in Grad zurück.

Argumente
x
numerisch, zwischen **-1** und **+1**

Resultat numerisch, liegt zwischen **0°** und **180°**

Erklärung Gibt den Arcus Cosinus des Eingabewerts in Grad zurück. Wenn der Eingabewert grösser als **+1** oder kleiner als **-1** ist, ist die Berechnung nicht möglich, und es wird ein Spezielles Symbol angezeigt.

Siehe auch **Cos(x)**, **Asin(x)**, **Atan(x)**, **Sin(x)**, **Tan(x)**, übrige Mathematik-Funktionen

Beispiel Siehe func_math2.qdf

5.3.3 addDay(date_value, number)

Kategorie Datums-Funktion

Kurzbeschreibung Berechnet das Datum, welches gegeüber dem Datum **date_value** um **number** Tage entfernt ist.

Argumente

date_value
gültiges Datum

number
ganze Zahl

Resultat Datum

Erklärung Gibt das Datum entsprechend der in **number** angegebenen Anzahl Tage ab dem Eingabedatum **date_value** zurück. Das Eingabedatum kann in der Zukunft oder Vergangenheit liegen. Eine positive Zahl **number** bedeutet ein Datum in der Zukunft gegenüber dem Eingabedatum, eine negative Zahl bedeutet ein Datum in der Vergangenheit.

Für Berechnungen ab dem aktuellen Datum kann als Wert für **date_value** die Funktion **Today()** benutzt werden. Wird für **number** eine Zahl mit Nachkommastellen eingegeben, werden diese ignoriert.

Zur Anzeige des Resultats ist ein entsprechend formatiertes Datumsfeld notwendig.

Siehe auch

addMonth(date_value, number), **addYear(date_value, number)**, **calcDays(date_first, date_second)**, **Date(year, month, day)**, **getDay(date_value)**, **Today()**, übrige Datums-Funktionen

Beispiel

Siehe func_date.qdf

5.3.4 addHour(time_value, number)

Kategorie

Datums-Funktion

Kurzbeschreibung

Berechnet die Uhrzeit, welche gegeüber der Uhrzeit **time_value** um **number** Stunden entfernt ist.

Argumente

time_value
gültige Uhrzeit

number
ganze Zahl

Resultat

Uhrzeit

Erklärung

Gibt die Uhrzeit entsprechend der in **number** angegebenen Anzahl Stunden ab der Eingabezeit **time_value** zurück. Die Eingabezeit kann in der Zukunft oder Vergangenheit liegen. Eine positive Zahl **number** bedeutet eine Uhrzeit in der Zukunft gegenüber der Eingabezeit, eine negative Zahl bedeutet eine Uhrzeit in der Vergangenheit.

Für Berechnungen ab der aktuellen Uhrzeit kann als Wert für **time_value** die Funktion **Now()** benutzt werden. Wird für **number** eine Zahl mit Nachkommastellen eingegeben, werden diese ignoriert.

In dieser Funktion werden nur Stunden addiert; der Wert für die Minuten bleibt unverändert. Gegebenenfalls muss der Minutenwert mit Hilfe der Funktion **addMinute(time_value, number)** verändert werden.

Zur Anzeige des Resultats ist ein entsprechend formatiertes Datumsfeld notwendig (Format **HH:mm**). Es ist möglich, das Datumsfeld als eigentliches Datum zu formatieren. In diesem Fall wird dann das zur berechneten Uhrzeit zugehörige Datum angezeigt.

Siehe auch

addMinute(time_value, number), **calcHours(date_first, date_second)**, **getHour(time_value)**, **Now()**, übrige Datums- und Uhrzeit-Funktionen.

Beispiel

Siehe func_time.qdf

5.3.5 addMinute(time_value, number)

Kategorie

Datums-Funktion

Kurzbeschreibung

Berechnet die Uhrzeit, welche gegeüber der Uhrzeit **time_value** um **number** Minuten entfernt ist.

Argumente

time_value
gültige Uhrzeit

number
ganze Zahl

Resultat

Uhrzeit

Erklärung

Gibt die Uhrzeit entsprechend der in **number** angegebenen Anzahl Minuten ab der Eingabezeit **time_value** zurück. Die Eingabezeit kann in der Zukunft oder Vergangenheit liegen. Eine positive Zahl **number** bedeutet eine Uhrzeit in der Zukunft gegenüber der Eingabezeit, eine negative Zahl bedeutet eine Uhrzeit in der Vergangenheit.

Für Berechnungen ab der aktuellen Uhrzeit kann als Wert für **time_value** die Funktion **Now()** benutzt werden. Wird für **number** eine Zahl mit Nachkommastellen eingegeben, werden diese ignoriert.

In dieser Funktion werden nur Minuten addiert; der Wert für die Stunden wird entsprechend berechnet.

Zur Anzeige des Resultats ist ein entsprechend formatiertes Datumsfeld notwendig (Format **HH:mm**). Es ist möglich, das Datumsfeld als eigentliches Datum zu formatieren. In diesem Fall wird dann das zur berechneten Uhrzeit zugehörige Datum angezeigt.

Siehe auch

addHour(time_value, number), **calcMinutes(date_first, date_second)**, **getMinute(time_value)**, **getMinute(time_value)**, **Now()**, übrige Datums- und Uhrzeit-Funktionen.

Beispiel

Siehe func_time.qdf

5.3.6 addMonth(date_value, number)

Kategorie

Datums-Funktion

Kurzbeschreibung

Berechnet das Datum, welches gegeüber dem Datum **date_value** um **number** Monate entfernt ist.

Argumente

date_value
gültiges Datum

number
ganze Zahl

Resultat

Datum

Erklärung

Gibt das Datum entsprechend der in **number** angegebenen Anzahl Monate ab dem Eingabedatum **date_value** zurück. Das Eingabedatum kann in der Zukunft oder Vergangenheit liegen. Eine positive Zahl **number** bedeutet ein Datum in der Zukunft gegenüber dem Eingabedatum, eine negative Zahl bedeutet ein Datum in der Vergangenheit.

Für Berechnungen ab dem aktuellen Datum kann als Wert für **date_value** die Funktion **Today()** benutzt werden. Wird für **number** eine Zahl mit Nachkommastellen eingegeben, werden diese ignoriert.

Zur Anzeige des Resultats ist ein entsprechend formatiertes Datumsfeld notwendig.

In dieser Funktion werden nur Monate addiert; der Wert für die Tage bleibt unverändert. Gegebenenfalls muss der Tagewert mit Hilfe der Funktion **addDay(date_value, number)** verändert werden.

Siehe auch

addDay(date_value, number), **addYear(date_value, number)**, **Date(year, month, day)**, **getMonth(date_value)**, **Today()**, übrige Datums-Funktionen

Beispiel

Siehe func_date.qdf

5.3.7 addYear(date_value, number)

Kategorie

Datums-Funktion

Kurzbeschreibung

Berechnet das Datum, welches gegeüber dem Datum **date_value** um **number** Jahre entfernt ist.

Argumente

date_value
gültiges Datum

number
ganze Zahl

Resultat

Datum

Erklärung

Gibt das Datum entsprechend der in **number** angegebenen Anzahl Jahre ab dem Eingabedatum **date_value** zurück. Das Eingabedatum kann in der Zukunft oder Vergangenheit liegen. Eine positive Zahl **number** bedeutet ein Datum in der Zukunft gegenüber dem Eingabedatum, eine negative Zahl bedeutet ein Datum in der Vergangenheit.

Für Berechnungen ab dem aktuellen Datum kann als Wert für **date_value** die Funktion **Today()** benutzt werden. Wird für **number** eine Zahl mit Nachkommastellen eingegeben, werden diese ignoriert.

Zur Anzeige des Resultats ist ein entsprechend formatiertes Datumsfeld notwendig.

In dieser Funktion werden nur Jahre addiert; der Wert für die Tage und Monate bleibt unverändert. Gegebenenfalls muss der Tage- und Monatewert mit Hilfe der Funktion **addDay(date_value, number)** bzw. **addMonth(date_value, number)** verändert werden.

Siehe auch

addDay(date_value, number), **addMonth(date_value, number)**, **Date(year, month, day)**, **getYear(date_value)**, **Today()**, übrige Datums-Funktionen

Beispiel

Siehe func_date.qdf

5.3.8 Asin(x)

Kategorie

Mathematik-Funktion

Kurzbeschreibung

Gibt den Arcus Sinus des Eingabewerts in Grad zurück.

Argumente — **x**
numerisch, zwischen **-1** und **+1**

Resultat — numerisch, liegt zwischen **-90°** und **+90°**

Erklärung — Gibt den Arcus Sinus des Eingabewerts in Grad zurück. Wenn der Eingabewert grösser als **+1** oder kleiner als **-1** ist, ist die Berechnung nicht möglich, und es wird ein Spezielles Symbol angezeigt.

Siehe auch — **Sin(x)**, **Acos(x)**, **Atan(x)**, **Cos(x)**, **Tan(x)**, übrige Mathematik-Funktionen

Beispiel — Siehe func_math2.qdf

5.3.9 Atan(x)

Kategorie — Mathematik-Funktion

Kurzbeschreibung — Gibt den Arcus Tangens des Eingabewerts in Grad zurück.

Argumente — **x**
numerisch, beliebig

Resultat — numerisch, liegt zwischen **-90°** und **+90°**

Erklärung — Gibt den Arcus Tangens des Eingabewerts in Grad zurück.

Siehe auch — **Tan(x)**, **Acos(x)**, **Asin(x)**, **Cos(x)**, **Sin(x)**, übrige Mathematik-Funktionen

Beispiel — Siehe func_math2.qdf

5.3.10 calcDays(date_first, date_second)

Kategorie — Datums-Funktion

Kurzbeschreibung Berechnet die Anzahl Tage zwischen den beiden Eingabedaten.

Argumente

date_first
gültiges Datum

date_second
gültiges Datum

Resultat numerisch, ganze Zahl

Erklärung Gibt die Anzahl Tage zwischen den beiden Daten **date_first** und **date_second** zurück.

Die Eingabedaten können in der Zukunft oder Vergangenheit liegen. Eine positives Resultat bedeutet, dass **date_second** später als **date_first** ist, ein negatives Resultat bedeutet, dass **date_second** früher als **date_first** ist.

Für Berechnungen mit dem aktuellen Datum kann als Wert für **date_first** bzw. date_second die Funktion **Today()** benutzt werden.

Siehe auch **calcHours(date_first, date_second)**, **calcMinutes(date_first, date_second)**, **addDay(date_value, number)**, **Date(year, month, day)**, **getDay(date_value)**, übrige Datums-Funktionen.

Beispiel Siehe func_date.qdf

5.3.11 calcHours(date_first, date_second)

Kategorie Datums-Funktion

Kurzbeschreibung Berechnet die Anzahl Stunden zwischen den beiden Eingabedaten.

Argumente

date_first
gültiges Datum und/oder Uhrzeit

date_second
gültiges Datum und/oder Uhrzeit

Resultat

numerisch, ganze Zahl

Erklärung

Gibt die Anzahl Stunden zwischen den beiden Daten **date_first** und **date_second** zurück.

Die Eingabedaten können in der Zukunft oder Vergangenheit liegen. Eine positives Resultat bedeutet, dass **date_second** später als **date_first** ist, ein negatives Resultat bedeutet, dass **date_second** früher als **date_first** ist.

Für Berechnungen mit dem aktuellen Datum kann als Wert für **date_first** bzw. **date_second** die Funktion **Today()** benutzt werden.

Wenn die Eingabedaten als Datum formatiert sind, wird in beiden Fällen die gleiche Uhrzeit eingesetzt. Das Resultat ist in diesem Fall immer ein Vielfaches von 24. Wenn die Eingabedaten Uhrzeitinformationen enthalten, werden die tatsächlich dazwischenliegenden Stunden berücksichtigt.

Siehe auch

calcDays(date_first, date_second), **calcHours(date_first, date_second)**, **addMinute(time_value, number)**, **Date(year, month, day)**, **getMinute(time_value)**, übrige Datums-Funktionen

Beispiel

Siehe func_time.qdf

5.3.12 calcMinutes(date_first, date_second)

Kategorie

Datums-Funktion

Kurzbeschreibung

Berechnet die Anzahl Minuten zwischen den beiden Eingabedaten.

Argumente

date_first
gültiges Datum und/oder Uhrzeit

date_second
gültiges Datum und/oder Uhrzeit

Resultat

numerisch, ganze Zahl

Erklärung

Gibt die Anzahl Minuten zwischen den beiden Daten **date_first** und **date_second** zurück.

Die Eingabedaten können in der Zukunft oder Vergangenheit liegen. Eine positives Resultat bedeutet, dass **date_second** später als **date_first** ist, ein negatives Resultat bedeutet, dass **date_second** früher als **date_first** ist.

Für Berechnungen mit dem aktuellen Datum kann als Wert für **date_first** bzw. **date_second** die Funktion **Today()** benutzt werden.

Wenn die Eingabedaten als Datum formatiert sind, wird in beiden Fällen die gleiche Uhrzeit eingesetzt. Das Resultat ist in diesem Fall immer ein Vielfaches von 24. Wenn die Eingabedaten Uhrzeitinformationen enthalten, werden die tatsächlich dazwischenliegenden Stunden berücksichtigt.

Siehe auch

calcDays(date_first, date_second), **calcHours(date_first, date_second)**, **addHour(time_value, number)**, **Date(year, month, day)**, **getHour(time_value)**, übrige Datums-Funktionen

Beispiel

Siehe func_time.qdf

5.3.13 Ceil(x)

Kategorie

Mathematik-Funktion

Kurzbeschreibung Rundet den Eingabewert auf die nächstgrössere ganze Zahl auf.

Argumente

x
numerisch, beliebiger Wert

Resultat

numerisch

Erklärung Gibt den auf die nächstgrössere ganze Zahl aufgerundeten Eingabewert zurück

Siehe auch **Cover(x)**, **Round(x)**, **Floor(x)**, **Trunc(x)**

Beispiel Siehe func_math1.qdf

5.3.14 Cell(table, column, row)

Kategorie Tabellen-Funktion

Kurzbeschreibung Stellt den Wert der Tabellenzelle von Tabelle **table**, Spalte **column**, Zeile **row** dar.

Argumente

table
Gültiger Name einer Tabelle

column
Gültiger Elementname oder Spaltennummer der Tabelle **table**

row
Ganze Zahl

Resultat Variabel, entsprechend dem Format des Elements für Spalte **column**

Erklärung

Gibt den Wert der Tabellenzelle von Tabelle table, Spalte column, Zeile row zurück.

Die Spalte **column** kann entweder über ihre interne Nummer (siehe Abschnitt 4.4.7.2) oder über den Namen des definierenden Felds angesprochen werden. Wenn dies keine gültige Spalte betrifft (z.B. Feld nicht Bestandteil der Tabelle oder Nummer grösser als totale Anzahl Spalten der Tabelle), wird **null** (entsprechend einem leeren Wert) zurückgegeben. Dasselbe gilt sinngemäss auch wenn der Wert von **row** grösser ist als die effektive Anzahl Zeilen der Tabelle.

Da der Typ des zurückgegebenen Werts nicht vorhersehbar ist, ist beim Festlegen der Formel allenfalls sicherzustellen, dass das Resultat, bzw. dessen Format auch tatsächlich weiter verarbeitet werden kann.

Siehe auch

getCol(), **getRow()**, **getValues(delimiter, value1 [, value2[, …]])**

Beispiel

Siehe func_table2.qdf

5.3.15 checkCardNumber(value)

Kategorie

Konsistenzprüfungs-Funktion

Kurzbeschreibung

Validiert Kreditkartennummern.

Argumente

value
Zahl oder String, entsprechend einer Kreditkartennummer

Resultat

boolean (**TRUE** oder **FALSE**)

Erklärung

Validiert die Richtigkeit von Kreditartennummern unter Verwendung des *Luhn-Algorithmus*, auch als "*MOD-10-Algorithmus*" bekannt. Bei erfolgreicher Prüfung ist das Resultat **TRUE**, anderenfalls ist es **FALSE**. Wenn der Eingabewert nicht numerisch oder ein String ist, ist das Resultat **NULL**.

Siehe auch **checkMod10(value)**, **getMod10(value)**

Beispiel Siehe func_consist.qdf

5.3.16 checkMod10(value)

Kategorie Konsistenzprüfungs-Funktion

Kurzbeschreibung Validiert Nummern unter Verwendung des "MOD-10-Algorithmus".

Argumente **value**
Zahl oder String, darf nur Ziffern und Leerzeichen enthalten

Resultat boolean (**TRUE** oder **FALSE**)

Erklärung Validiert die Richtigkeit von Nummern unter Verwendung des "*MOD-10-Algorithmus*". Bei erfolgreicher Prüfung ist das Resultat **TRUE**, anderenfalls ist es **FALSE**. Wenn der Eingabewert nicht numerisch oder ein String ist, ist das Resultat **NULL**.

Siehe auch **checkCardNumber(value)**, **getMod10(value)**, **checkMod10r(value)**

Beispiel Siehe func_consist.qdf

5.3.17 checkMod10r(value)

Kategorie Konsistenzprüfungs-Funktion

Kurzbeschreibung Validiert Nummern unter Verwendung des "rekursiven MOD-10-Algorithmus".

Argumente **value**
Zahl oder String, darf nur Ziffern und Leerzeichen enthalten

Resultat boolean (**TRUE** oder **FALSE**)

Erklärung Validiert die Richtigkeit von Nummern unter Verwendung des "*rekursiven MOD-10-Algorithmus*". Bei erfolgreicher Prüfung ist das Resultat **TRUE**, anderenfalls ist es **FALSE**. Wenn der Eingabewert nicht numerisch oder ein String ist, ist das Resultat **NULL**.

Siehe auch **checkMod10(value)**, **checkCardNumber(value)**, **getMod10(value)**

Beispiel Siehe func_consist.qdf

5.3.18 Compress(x)

Kategorie Spezial-Funktion

Kurzbeschreibung Erzeugt aus dem Eingabewert eine mit ZLIB komprimierte Bytefolge, welche in einem Barcode mit Codierungsmethode **Binär** dargestellt werden kann.

Argumente **x**
String, Bytefolge; numerische Werte werden automatisch in einen String umgewandelt

Resultat Bytefolge

Erklärung Erzeugt aus dem Eingabwert eine komprimierte Bytefolge, welche in einem 2D-Barcode mit Methode **Binär** codiert werden kann. Diese Funktion wird praktisch nur im Zusammenhang mit 2D-Barcodes eingesetzt.

Strings werden direkt komprimiert. Numerische Werte werden intern in einen String umgewandelt und komprimiert. Datumswerte müssen vorgängig in einen String umgewandelt werden (mit der Funktion **Str(x)**).

Siehe auch **getBytes(string)**, **SubArray(byte_array, start [, count])**

Beispiel Siehe func_barcode.qdf

5.3.19 Cos(x)

Kategorie Mathematik-Funktion

Kurzbeschreibung Berechnet den Cosinus eines in Grad definierten Winkels.

Argumente **x**
numerisch, beliebiger Wert, die Eingabewerte wiederholen sich in Zyklen von 360°

Resultat numerisch, liegt zwischen **-1** und **+1**

Erklärung Gibt den Cosinus des Eingabewerts zurück. Der Eingabewert wird in Grad genommen. Der volle Kreis entspricht 360°.

Siehe auch **Acos(x)**, **Asin(x)**, **Atan(x)**, **Sin(x)**, **Tan(x)**, übrige Mathematik-Funktionen

Beispiel Siehe func_math2.qdf

5.3.20 Cover(x)

Kategorie Mathematik-Funktion

Kurzbeschreibung Rundet den Absolutwert des Eingabewerts auf die nächstgrössere ganze Zahl auf.

Argumente **x**
numerisch, beliebiger Wert

Resultat numerisch

Erklärung Gibt, ohne auf das Vorzeichen zu achten, den auf die nächstgrössere ganze Zahl aufgerundeten Eingabewert zurück. Diese Funktion entspricht der Gleichung

$$\mathrm{Cover(x)} = \frac{\mathrm{x}}{\mathrm{Abs(x)}} \times \mathrm{Ceil(x)}$$

Diese Funktion ist daher für positive Zahlen identisch mit **Ceil(x)**; für negative Zahlen weicht der Zahlenwert gegenüber von **Ceil(x)** um 1 ab.

Siehe auch **Ceil(x)**, **Round(x)**, **Floor(x)**, **Trunc(x)**

Beispiel Siehe func_math1.qdf

5.3.21 createGUID()

Kategorie Identifikations-Funktion

Kurzbeschreibung Erzeugt einen einmaligen Identifikationsstring (GUID, Global Unique Identifier).

Argumente Diese Funktion hat keine Argumente.

Resultat String

Erklärung Erzeugt eine aus 36 Zeichen bestehende strukturierte Zeichenkette, welche aus zufällig ausgewählten Zeichen besteht. Der hierfür eingesetzte Algorithmus stellt sicher, dass die Zeichenkette weltweit einmalig ist (im Vergleich mit anderen auf diese Art und weise erzeugten Zeichenketten).

Siehe auch **getRandomString(length [, digits])**, **getSessionID()**, **getPrintID()**, **getDataID()**, **getPrintIdStr(length [, digits]**

Beispiel Siehe func_ident.qdf

5.3.22 Date(year, month, day)

Kategorie Datum-Funktion

Kurzbeschreibung Erzeugt ein Datumsobjekt aus den Argumenten.

Argumente

year
Ganze positive Zahl, entsprechend der Jahrzahl

month
Ganze positive Zahl, entsprechend der Monatszahl

day
Ganze positive Zahl, entsprechend der Tageszahl

Resultat Datum

Erklärung Erzeugt ein Datum aus den drei Argumenten. Die Eingabe für **year** muss für heutige Daten vierstellig erfolgen (die Eingabe von zweistelligen Zahlen führt zu Daten aus der Römerzeit).

Der Wert für **month** sollte zwischen 1 und 12 liegen. Grössere Werte werden umgerechnet und die Jahrzahl entsprechend erhöht (Beispiel: entspricht dem März des folgenden Jahres).

Der Wert für **day** sollte zwischen 1 und 28 bis 31, je nach Wert für **month**. Grössere Werte werden umgerechnet und entsprechend dem Monat und Jahr zugeschlagen (Beispiel: der “35. Mai” entspricht dem 4. Juni).

Die Eingaben sollten ganze Zahlen sein, wobei trotzdem eingegebene Nachkommastellen ignoriert werden.

Siehe auch — **getDay(date_value)**, **getMonth(date_value)**, **getYear(date_value)**, sowie übrige Datum-Funktionen

Beispiel — Siehe func_date.qdf

5.3.23 Exp(x)

Kategorie — Mathematik-Funktion

Kurzbeschreibung — Berechnet den Wert der Exponentialfunktion e^x.

Argumente — **x**
numerisch, beliebiger Wert

Resultat — numerisch, grösser als 0

Erklärung — Berechnet den Wert der Exponentialfunktion e^x für den Eingabewert und gibt das Resultat zurück. Bei Eingabewerten grösser als **0** ist das Resultat grösser als **1**; kleiner als **0** liegt es zwischen **0** und **1**. e^0 hat den Wert **1**. Die Umkehrfunktion ist **Log(x)**.

Siehe auch — **Log(x)**

Beispiel — Siehe func_math2.qdf

5.3.24 find(string, regex[, start])

Kategorie — Regular Expression-Funktion

Kurzbeschreibung — Findet die Position der ersten nach Position **start** vorkommenden Fundstelle des in **regex** definierten Musters im Eingabewert **string**.

Argumente

string
String

regex
Regular Expression

start (optional)
Positionsnummer, ganze Zahl beginnend mit 0

Resultat

numerisch, ganze Zahl zwischen 0 und der Länge von **string** - 1, oder **-1**

Erklärung

Findet die Positionsnummer der ersten nach der Position **start** vorkommenden Fundstelle des in **regex** definierten Musters im Eingabewert **string**. Das Resultat ist die Positionsnummer oder, wenn keine Fundstelle vorhanden ist, **-1**. Die Nummerierung der Positionen beginnt mit 0. Wenn das Argument **start** fehlt, wird mit Position 0 begonnen.

Eine Übersicht über die Syntax von Regular Expressions befindet sich in Abschnitt 5.4.

Siehe auch

match(string, regex), **replaceAll(orig_string, regex, replace_with)**

Beispiel

Siehe func_regex.qdf

5.3.25 Floor(x)

Kategorie

Mathematik-Funktion

Kurzbeschreibung

Rundet den Eingabewert auf die nächstkleinere ganze Zahl ab.

Argumente

x
numerisch, beliebiger Wert

Resultat

numerisch

Erklärung

Gibt den auf die nächstkleinere ganze Zahl abgerundeten Eingabewert zurück. Dies entspricht im Wesentlichen dem Abschneiden der Nachkommastellen (zumindest bei positiven Zahlen).

Siehe auch

Trunc(x), **Round(x)**, **Ceil(x)**, **Cover(x)**

Beispiel

Siehe func_math1.qdf

5.3.26 getBytes(string)

Kategorie

String-Funktion

Kurzbeschreibung

Codiert die eingegebene Zeichenkette zu einer Bytefolge, welche in einem 2D-Barcode mit Codierung **Binär** eingegeben werden kann.

Argumente

string
numerisch, beliebiger Wert

Resultat

Bytefolge

Erklärung

Wandelt den als Argument übergebenen String zu einer Bytefolge um, welche zur Erzeugung eines 2D-Barcodes mit Codierung **Binär** dienen kann. Für die Umwandlung wird die UTF-8-Zeichensatzcodierung verwendet.

Die mit dieser Funktion erzeugte Bytefolge kann mit der Funktion **Compress(x)** weiter komprimiert werden.

Hinweis: *Die in früheren Snapform-Versionen verwendete ISO 8859-1-Zeichensatzcodierung ist mit UTF-8 kompatibel, so dass sich keine Versionsprobleme ergeben.*

Siehe auch

Compress(x), **SubArray(byte_array, start [, count])**

Beispiel Siehe func_string2.qdf und func_barcode.qdf

5.3.27 getCol()

Kategorie Tabellen-Funktion

Kurzbeschreibung Gibt die Spaltennummer des betreffenden Felds in der zugehörigen Tabelle an.

Argumente Diese Funktion hat keine Argumente.

Resultat numerisch, ganze Zahl

Erklärung Gibt die Nummer der Spalte in der Tabelle des aufrufenden Felds zurück. Dieser Wert kann in nachfolgenden Berechnungen weiter verwendet werden.

Der Aufruf dieser Funktion ist nur im Zusammenhang mit Tabellen sinnvoll. Wenn das aufrufende Feld nicht Bestandteil einer Tabelle ist, gibt diese Funktion den Wert **-1** zurück.

Siehe auch **getRow()**, **Cell(table, column, row)**

Beispiel Siehe func_table2.qdf

5.3.28 getContent(url)

Kategorie Spezial-Funktion

Kurzbeschreibung Lädt den (Text)-Inhalt von **url**.

Argumente **url**
gültiger URL

Resultat

Der HTML 3.2-kompatible Teil von URL wird geladen

Erklärung

Diese Funktion dient zum Laden von URL-Inhalten. Diese Inhalte sind über einen gültigen Pfad **url** abzurufen. Es werden im Wesentlichen alle URI-Protokolle unterstützt (http:, https:, file:, ftp:, etc.).

Die Inhalte müssen HTML 3.2-kompatibel sein. Die primäre Bedeutung dieser Funktion ist das Laden von Texten in Textfelder, Infopunkte oder Beschriftungsfelder; kurz gesagt, überall dort, wo HTML-formatierter Text angezeigt werden kann. Eine weitere Bedeutung ist das Nutzen von Webservices, sofern deren Antwort in einem geeigneten Format ausgegeben wird.

Siehe auch

importData(url), sowie Snapform-Datenstruktur

Beispiel

Siehe func_misc.qdf

5.3.29 getDataID()

Kategorie

Identifikations-Funktion

Kurzbeschreibung

Erzeugt einen einmaligen Identifikationsstring (GUID, Global Unique Identifier), welcher mit einem Satz Formulardaten verknüpft werden kann.

Argumente

Diese Funktion hat keine Argumente.

Resultat

String

Erklärung

Erzeugt eine aus 36 Zeichen bestehende strukturierte Zeichenkette, welche aus zufällig ausgewählten Zeichen besteht. Der hierfür eingesetzte Algorithmus stellt sicher, dass die Zeichenkette weltweit einmalig ist (im Vergleich mit anderen auf diese Art und weise erzeugten Zeichenketten).

Diese Identifikation kann zum Beispiel bei der Übertragung von Daten oder bei digitalen Arbeitsabläufen eingesetzt werden.

Siehe auch — **createGUID()**, **getSessionID()**, **getPrintID()**, **getPrintIdStr(length [, digits]**, **getRandomString(length [, digits])**

Beispiel — Siehe func_ident.qdf

5.3.30 getDay(date_value)

Kategorie — Datums-Funktion

Kurzbeschreibung — Gibt den Monatstag des Datums **date_value** an.

Argumente — **date_value**
gültiges Datum

Resultat — Ganze Zahl zwischen **1** und **31**

Erklärung — Gibt den Monatstag des Eingabedatum **date_value** zurück. Das Eingabedatum kann in der Zukunft oder Vergangenheit liegen.

Diese Funktion ist Teil der Umkehrfunktion von **Date(year, month, day)**.

Siehe auch — **getMonth(date_value)**, **getYear(date_value)**, **Date(year, month, day)**, übrige Datums-Funktionen

Beispiel — Siehe func_date.qdf

5.3.31 getHour(time_value)

Kategorie — Datums-Funktion

Kurzbeschreibung Gibt den Stundenwert der Uhrzeit **time_value** an.

Argumente **time_value**
gültige Uhrzeit

Resultat Ganze Zahl zwischen **0** und **24**

Erklärung Gibt den Stundenwert der Eingabe-Uhrzeit **time_value** zurück.

Diese Funktion ist Teil der Umkehrfunktion von **Time(hour, minute[, second])**.

Siehe auch **getMinute(time_value)**, **Time(hour, minute[, second])**, übrige Datums-Funktionen

Beispiel Siehe func_time.qdf

5.3.32 getMinute(time_value)

Kategorie Datums-Funktion

Kurzbeschreibung Gibt den Minutenwert der Uhrzeit **time_value** an.

Argumente **time_value**
gültige Uhrzeit

Resultat Ganze Zahl zwischen **0** und **59**

Erklärung Gibt den Minutenwert der Eingabe-Uhrzeit **time_value** zurück.

Diese Funktion ist Teil der Umkehrfunktion von **Time(hour, minute[, second])**.

Siehe auch — **getHour(time_value)**, **Time(hour, minute[, second])**, übrige Datums-Funktionen

Beispiel — Siehe func_time.qdf

5.3.33 getMod10(value)

Kategorie — Konsistenzprüfungs-Funktion

Kurzbeschreibung — Gibt eine mit Hilfe des “MOD-10-Algorithmus” erzeugte Prüfsumme über den Eingabwert **value** zurück.

Argumente — **value**
Zahl oder String, enthaltend numerische oder alphanumerische Zeichen

Resultat — Ganze positive Zahl

Erklärung — Errechnet eine Prüfsumme über den Eingabewert **value**, unter Anwendung des “*MOD-10-Algorithmus*”. Das Resultat ist eine ganze Zahl, welche beispielsweise in gewissen 1D-Barcodes weiterverwendet werden kann.

Siehe auch — **checkMod10(value)**, **getMod10r(value)**, **checkMod10r(value)**

Beispiel — Siehe func_consist.qdf

5.3.34 getMod10r(value)

Kategorie — Konsistenzprüfungs-Funktion

Kurzbeschreibung — Gibt eine mit Hilfe des “rekursiven MOD-10-Algorithmus” erzeugte Prüfsumme über den Eingabwert **value** zurück.

Argumente

value
Zahl oder String, enthaltend numerische oder alphanumerische Zeichen

Resultat

Ganze positive Zahl

Erklärung

Errechnet eine Prüfsumme über den Eingabewert **value**, unter Anwendung des "rekursiven *MOD-10-Algorithmus*". Das Resultat ist eine ganze Zahl, welche beispielsweise in gewissen 1D-Barcodes weiterverwendet werden kann.

Siehe auch

checkMod10r(value), **getMod10(value)**, **checkMod10(value)**

Beispiel

Siehe func_consist.qdf

5.3.35 getMonth(date_value)

Kategorie

Datums-Funktion

Kurzbeschreibung

Gibt den Monatswert des Datums **date_value** an.

Argumente

date_value
gültiges Datum

Resultat

Ganze Zahl zwischen **1** und **12**

Erklärung

Gibt den Monatswert des Eingabedatum **date_value** zurück. Das Eingabedatum kann in der Zukunft oder Vergangenheit liegen.

Diese Funktion ist Teil der Umkehrfunktion von **Date(year, month, day)**.

Siehe auch

getDay(date_value), **getYear(date_value)**, **Date(year, month, day)**, übrige Datums-Funktionen

Beispiel Siehe func_date.qdf

5.3.36 getPrintID()

Kategorie Identifikations-Funktion

Kurzbeschreibung Erzeugt einen einmaligen Identifikationsstring (GUID, Global Unique Identifier), welcher mit einem Druckauftrag verknüpft werden kann.

Argumente Diese Funktion hat keine Argumente.

Resultat String

Erklärung Erzeugt eine aus 36 Zeichen bestehende strukturierte Zeichenkette, welche aus zufällig ausgewählten Zeichen besteht. Der hierfür eingesetzte Algorithmus stellt sicher, dass die Zeichenkette weltweit einmalig ist (im Vergleich mit anderen auf diese Art und weise erzeugten Zeichenketten).

Diese Identifikation kann zum Beispiel bei Druckaufträgen eingesetzt werden.

Siehe auch **getPrintIdStr(length [, digits]**, **createGUID()**, **getSessionID()**, **getDataID()**, **getRandomString(length [, digits])**

Beispiel Siehe func_ident.qdf

5.3.37 getPrintIdStr(length [, digits]

Kategorie Identifikations-Funktion

Kurzbeschreibung Erzeugt einen frei konfigurierbaren einmaligen Identifikationsstring, welcher mit einem Druckauftrag verknüpft werden kann.

Argumente

length
Länge des zu erzeugenden Strings; positive ganze Zahl

digits (optional)
Angabe, ob der String rein numerisch sein soll; boolean, **FALSE** wenn nicht spezifiziert

Resultat

String, alphanumerisch oder numerisch (je nach Wert des Arguments **digits**)

Erklärung

Erzeugt einen einmaligen Identifikationsstring, welcher mit einem Druckauftrag verknüpft werden kann. Die Länge des Strings ist über das Argument **length** einstellbar. Das Argument **digits** legt fest, ob der String rein numerisch sein soll.

Siehe auch

getPrintID(), **createGUID()**, **getSessionID()**, **getDataID()**, **getRandomString(length [, digits])**

Beispiel

Siehe func_ident.qdf

5.3.38 getRandomString(length [, digits])

Kategorie

Identifikations-Funktion

Kurzbeschreibung

Erzeugt einen zufällig aufgebauten String der Länge **length**.

Argumente

length
Länge des zu erzeugenden Strings; positive ganze Zahl

digits (optional)
Angabe, ob der String rein numerisch sein soll; boolean, **FALSE** wenn nicht spezifiziert

Resultat

String, alphanumerisch oder numerisch (je nach Wert des Arguments **digits**)

Erklärung

Erzeugt einen zufällig aufgebauten String. Die Länge des Strings ist über das Argument **length** einstellbar. Das Argument **digits** legt fest, ob der String rein numerisch sein soll. Diese Funktion wird auch eingesetzt (mit dem Wert **digits** gleich **TRUE**), wenn eine Zufallszahl erzeugt werden soll.

Siehe auch

getPrintIdStr(length [, digits], **createGUID()**, **getSessionID()**,**getPrintID()**, **getDataID()**

Beispiel

Siehe func_ident.qdf

5.3.39 getRow()

Kategorie

Tabellen-Funktion

Kurzbeschreibung

Gibt die Zeilennummer des betreffenden Felds in der zugehörigen Tabelle an.

Argumente

Diese Funktion hat keine Argumente.

Resultat

numerisch, ganze Zahl

Erklärung

Gibt die Nummer der Zeile in der Tabelle des aufrufenden Felds zurück. Dieser Wert kann in nachfolgenden Berechnungen weiter verwendet werden.

Der Aufruf dieser Funktion ist nur im Zusammenhang mit Tabellen sinnvoll. Wenn das aufrufende Feld nicht Bestandteil einer Tabelle ist, gibt diese Funktion den Wert **-1** zurück.

Siehe auch

getCol(), **Cell(table, column, row)**

Beispiel

Siehe func_table2.qdf

5.3.40 getSelectedIndex(ComboBox)

Kategorie
Spezial-Funktion

Kurzbeschreibung
Gibt den Index der aktuellen Auswahl eines Auswahllistenfelds zurück.

Argumente
ComboBox
Name eines Auswahllistenfelds

Resultat
numerisch, ganze Zahl zwischen 1 und der Anzahl Einträge des Auswahllistenfelds

Erklärung
Gibt den Index der aktuellen Auswahl eines Auswahllistenfelds zurück. Die Numerierung beginnt mit 1.

Siehe auch
setList(ComboBox, str_items, str_values), **setSelectedIndex(ComboBox, index)**

Beispiel
Siehe func_combobox.qdf

5.3.41 getSessionID()

Kategorie
Identifikations-Funktion

Kurzbeschreibung
Erzeugt einen einmaligen Identifikationsstring (GUID, Global Unique Identifier), welcher der laufenden Arbeitssitzung mit Snapform Viewer zugewiesen werden kann.

Argumente
Diese Funktion hat keine Argumente.

Resultat
String

Erklärung

Erzeugt eine aus 36 Zeichen bestehende strukturierte Zeichenkette, welche aus zufällig ausgewählten Zeichen besteht. Der hierfür eingesetzte Algorithmus stellt sicher, dass die Zeichenkette weltweit einmalig ist (im Vergleich mit anderen auf diese Art und weise erzeugten Zeichenketten).

Diese Identifikation dient zur Kennzeichnung der Arbeitssitzung mit Snapform Viewer und bleibt während dieser Sitzung konstant.

Siehe auch

createGUID(), **getPrintID()**, **getDataID()**, **getRandomString(length [, digits])**

Beispiel

Siehe func_ident.qdf

5.3.42 getValues(delimiter, value1 [, value2[, …]])

Kategorie

Tabellen-Funktion

Kurzbeschreibung

Fügt die Werte der angegebenen Felder zu einem String zusammen und benutzt dabei das Trennzeichen **delimiter**.

Argumente

delimiter
Trennzeichen, muss zwischen Anführungszeichen eingegeben werden.

value1 … valuen
Bezeichnung der Felder, deren Werte im Resultat zusammengefügt werden sollen

Resultat

String

Erklärung

Fügt die Werte der als **value1 … valuen** angegebenen Feldbezeichnungen zu einem String zusammen. Das Trennzeichen ist **delimiter**. Diese Funktion dient hauptsächlich zum Aufbereiten von Datenströmen für 2D-Barcodes, kann aber auch für die Datenübertragung eingesetzt werden.

Die Feldbezeichnungen dürfen nicht Bestandteil einer Tabelle sein. Bei Feldwerten, welche Tabellen-Elementen zugewiesen sind, werden keine Daten in das Resultat eingefügt. Zum Auslesen von Tabellenwerten wird die Funktion **TblValues(table, tbl_start, tbl_end, row_start, row_end, delimiter, column [, column[, …]])** benutzt.

Feldwerte werden als Strings dargestellt, auch wenn sie ursprünglich numerisch oder als Datum definiert worden sind.

Siehe auch **TblValues(table, tbl_start, tbl_end, row_start, row_end, delimiter, column [, column[, …]])**

Beispiel Siehe func_table2.qdf

5.3.43 getVersion()

Kategorie Spezial-Funktion

Kurzbeschreibung Gibt die Versionsnummer der benutzten Snapform-Applikation (Designer, Viewer) zurück.

Argumente Diese Funktion hat keine Argumente.

Resultat String

Erklärung Gibt die Versionsnummer der benutzten Snapform-Applikation (Designer, Viewer) zurück. Diese Nummer kann zum Abfangen von allfälligen Kompatibilitätsproblemen zwischen Snapform-Versionen verwendet werden.

Siehe auch keine verwandten Funktionen

Beispiel Siehe func_ident.qdf

5.3.44 getYear(date_value)

Kategorie

Datums-Funktion

Kurzbeschreibung

Gibt die Jahrzahl des Datums **date_value** an.

Argumente

date_value
gültiges Datum

Resultat

Ganze Zahl

Erklärung

Gibt die Jahrzahl des Eingabedatum **date_value** zurück. Das Eingabedatum kann in der Zukunft oder Vergangenheit liegen.

Diese Funktion ist Teil der Umkehrfunktion von **Date(year, month, day)**.

Siehe auch

getDay(date_value), **getMonth(date_value)**, **Date(year, month, day)**, übrige Datums-Funktionen

Beispiel

Siehe func_date.qdf

5.3.45 if(logical_test, value_if_true [,value_if_false](

Kategorie

Spezial-Funktion

Kurzbeschreibung

Führt eine Vergleichs-Operation auf den darin angegebenen Operanden durch und gibt je nach Resultat einen entsprechenden Wert zurück.

Argumente

logical test
Vergleichs-Operation

value_if_true
Rückgabewert wenn die Vergleichs.Operation das Resultat **TRUE**

zurückgibt; das Format des Rückgabewerts wird durch den Wert selbst bestimmt.

value_if_false
Rückgabewert wenn die Vergleichs.Operation das Resultat **FALSE** zurückgibt; das Format des Rückgabewerts wird durch den Wert selbst bestimmt.

Resultat

numerisch

Erklärung

Diese Funktion führt eine Vergleichs-Operation auf zwei Operanden (welche auch Formeln sein können) durch und gibt beim Resultat **TRUE** den Wert **value_if_true** zurück. Beim Resultat **FALSE** gibt sie den Wert **value_if_false** zurück. Diese Rückgabewerte können ihrerseits Formeln sein.

Die Art der beiden Rückgabewerte **value_if_true** und **value_if_false** sollte gleich sein (z.B. String oder numerisch oder boolean), um allfällige Konflikte mit Feldtypen zu verhindern.

Siehe auch

Vergleichs-Operatoren (Abschnitt 5.2.7 bis Abschnitt 5.2.14)

Beispiel

Siehe func_misc.qdf

5.3.46 importData(url)

Kategorie

Spezial-Funktion

Kurzbeschreibung

Importiert eine Datendatei von **url**.

Argumente

url
gültiger URL

Resultat

Die in der importierten Datei definierten Feldnamen-Feldwerte-Paare werden geladen

Erklärung

Diese Funktion dient zum Importieren von extern gespeicherten Daten. Die Dateien sind über einen gültigen Pfad **url** abzurufen. Es werden im Wesentlichen alle URI-Protokolle unterstützt (http:, https:, file:, ftp:, etc.).

Die Importdateien müssen entweder im QDF/A-Format oder als XML mit der Snapform-Datenstruktur vorliegen.

Feldnamen-Feldwerte-Paare in der Importdatei werden in die betreffenden Felder des Formulars geladen. Nicht im Formular vorhandene Feldnamen/Feldwerte-Paare werden ignoriert. Dasselbe gilt für nicht den Eingabeformaten entsprechende Dateien.

Siehe auch

getContent(url), sowie Snapform-Datenstruktur

Beispiel

Siehe func_misc.qdf

5.3.47 IncVersion(initial_value, step)

Kategorie

Mathematik-Funktion

Kurzbeschreibung

Schleifenzähler: setzt einen Zähler auf **inital_value** und erhöht ihn bei jedem folgenden Aufruf um **step**.

Argumente

initial_value
numerisch

step
numerisch

Resultat

numerisch

Erklärung

Diese Funktion gibt beim ersten Aufruf den Anfangswert **initial_value** zurück und hält diesen gespeichert. Bei jedem nachfolgenden Aufruf wird dieser Wert um **step** erhöht, gespeichert und zurückgegeben.

Diese Funktion kann als Schleifenzähler eingesetzt werden. In diesem Fall sind beide Argumente ganze Zahlen.

Siehe auch

keine verwandten Funktionen vorhanden

Beispiel

Siehe func_math1.qdf

5.3.48 InStr(string, pattern [, start_position])

Kategorie

String-Funktion

Kurzbeschreibung

Gibt die Position von **pattern** im **string**, beginnend mit der Suche ab Position **start_position** zurück.

Argumente

string
zu durchsuchender String oder Feldbezeichnung

pattern
String nach dem gesucht werden soll

start_position
Position im String string, ab welcher gesucht werden soll; optional, ganze Zahl grösser als oder gleich **0**; Grundwert wenn nicht angegeben ist **0**.

Resultat

numerisch, ganze Zahl grösser als oder gleich **0**; **-1** wenn **pattern** nicht in **string** enthalten

Erklärung

Sucht den String **string** nach **pattern** ab und gibt die Position, an welcher **pattern** beginnt zurück.

Wenn für **pattern** ein String eingegeben wird, muss dieser zwischen Anführungszeichen gesetzt werden.

Mit **start_position** kann angegeben werden, ab welcher Position die Suche beginnen soll. Dies kann praktisch sein, wenn nicht nach dem ersten Vorkommen von **pattern** gesucht werden soll.

Die Positionsnummern sind 0-basiert. Daher ist die Position des ersten Zeichens **0**. Wenn **pattern** nicht vorkommt, wird **-1** zurückgegeben.

Die Suche ist exakt, was bedeutet, dass Gross- und Kleinschreibung berücksichtigt wird. Dies ist bei der Ausgestaltung der Formel in Betracht zu ziehen.

Siehe auch

Left(string, count), **Right(string, count)**, **SubStr(string, start[, count])** sowie die übrigen String-Funktionen

Beispiel

Siehe func_string2.qdf

5.3.49 isEmpty(x)

Kategorie

Spezial-Funktion

Kurzbeschreibung

Prüft ob **x** der Leere Wert (bzw. **null**) ist.

Argumente

x
Feldbezeichnung oder Resultat einer weiteren Berechnung

Resultat

boolean; **TRUE** wenn **x** leerer Wert, sonst **FALSE**

Erklärung

Prüft, ob das Argument den Leeren Wert (bzw. null) enthält. Dies ist beispielsweise sinnvoll bei einer Vollständigkeitskontrolle des Formulars, oder zum Abfangen von falschen Resultaten, wenn unzulässige Werte vorhanden sein können (z.B. wenn eine Division durch 0 möglich ist).

Siehe auch keine direkt verwandten Funktionen vorhanden

Beispiel Siehe func_misc.qdf

5.3.50 Left(string, count)

Kategorie String-Funktion

Kurzbeschreibung Gibt die ersten **count** Zeichen, von links her gezählt, des Strings **string** zurück.

Argumente

string
String oder Feldname eines Textfelds

count
Anzahl zurückzugebender Zeichen; positive ganze Zahl

Resultat String oder Leerer Wert (wenn **string** weniger Zeichen als **count** aufweist)

Erklärung Gibt die ersten **count** Zeichen des Strings **string** zurück. Dabei wird von links her gezählt.

Diese Funktion gibt nur dann einen gültigen Wert zurück, wenn **string** mindestens **count** Zeichen lang ist. Wenn **string** kürzer ist, wird der Leere Wert zurückgegeben.

Siehe auch **Right(string, count)**, **SubStr(string, start[, count])** und übrige Text-Funktionen

Beispiel Siehe func_string1.qdf

5.3.51 Len(x)

Kategorie
String-Funktion

Kurzbeschreibung
Gibt die Länge (Anzahl Zeichen) eines Strings zurück.

Argumente
x
String

Resultat
numerisch; positive ganze Zahl oder Leerer Wert (wenn **x** nicht ein String ist)

Erklärung
Gibt die Länge (Anzahl Zeichen) des Strings **x** zurück.

Siehe auch
Str(x), **InStr(string, pattern [, start_position])**, **Left(string, count)**, **Right(string, count)** und übrige String-Funktionen

Beispiel
Siehe func_string1.qdf

5.3.52 Log(x)

Kategorie
Mathematik-Funktion

Kurzbeschreibung
Berechnet den natürlichen Logarithmus von x (ln x).

Argumente
x
numerisch, positive Zahl

Resultat
numerisch

Erklärung
Berechnet den natürlichen Logarithmus (Basis **e**) von **x** und gibt diesen zurück.

x muss grösser als **0** sein. Wenn **x** gleich **0** ist das Resultat -∞. Wenn **x** kleiner als **0** ist, ist der Logarithmus nicht definiert, und es wird die Fehlermarke angezeigt.

Häufig kommt auch der Logarithmus mit Basis 10 vor. Dieser wird folgendermassen berechnet:

```
= Log(x)/Log(10)
```

und analog der Logarithmus mit Basis 2:

```
= Log(x)/Log(2)
```

Die Umkehrfunktion ist **Exp(x)**.

Siehe auch

Exp(x)

Beispiel

Siehe func_math2.qdf

5.3.53 match(string, regex)

Kategorie

Regular Expression-Funktion

Kurzbeschreibung

Stellt fest, ob das in **regex** definierte Muster im Eingabewert **string** enthalten ist.

Argumente

string
String

regex
Regular Expression

Resultat

boolean (**TRUE** oder **FALSE**)

Erklärung

Stellt fest, ob der Eingabewert string eine Fundstelle für das im Regular Expression regex definierte Muster enthält.

Eine Übersicht über die Syntax von Regular Expressions befindet sich in Abschnitt 5.4.

Siehe auch **find(string, regex[, start])**, **replaceAll(orig_string, regex, replace_with)**

Beispiel Siehe func_regex.qdf

5.3.54 Max(x, y)

Kategorie Mathematik-Funktion

Kurzbeschreibung Bestimmt den grösseren der beiden Werte.

Argumente

x
numerisch, beliebiger Wert

y
numerisch, beliebiger Wert

Resultat numerisch

Erklärung Bestimmt den grösseren der beiden Werte **x** und **y** und gibt diesen zurück. Wenn beide Werte gleich sind, wird dieser zurückgegeben.

Siehe auch **Min(x, y)**

Beispiel Siehe func_math1.qdf

5.3.55 Min(x, y)

Kategorie Mathematik-Funktion

Kurzbeschreibung Bestimmt den kleineren der beiden Werte.

Argumente

x
numerisch, beliebiger Wert

y
numerisch, beliebiger Wert

Resultat numerisch

Erklärung Bestimmt den kleineren der beiden Werte **x** und **y** und gibt diesen zurück. Wenn beide Werte gleich sind, wird dieser zurückgegeben.

Siehe auch **Max(x, y)**

5.3.56 Now()

Kategorie Datums-Funktion

Kurzbeschreibung Gibt die aktuelle Uhrzeit zurück.

Argumente Diese Funktion hat keine Argumente.

Resultat Uhrzeit

Erklärung Gibt die aktuelle Uhrzeit zurück. Um diese darzustellen ist ein als Uhrzeit formatiertes Datumsfeld notwendig.

Siehe auch **getHour(time_value)**, **getMinute(time_value)** und die übrigen Uhrzeit-orientierten Datums-Funktionen

Beispiel Siehe func_math1.qdf

5.3.57 replaceAll(orig_string, regex, replace_with)

Kategorie

Regular Expression-Funktion

Kurzbeschreibung

Ersetzt sämtliche Fundstellen des in **regex** definierten Musters im Eingabewert **orig_string** durch den String **replace_with**.

Argumente

orig_string
String

regex
Regular Expression

replace_with
String

Resultat

String

Erklärung

Ersetzt sämtliche Fundstellen des in **regex** definierten Musters im Eingabewert **orig_string** durch den String **replace_with**. Wenn keine Fundstelle vorhanden ist, wird der Eingabewert zurückgegeben.

Eine Übersicht über die Syntax von Regular Expressions befindet sich in Abschnitt 5.4.

Siehe auch

find(string, regex[, start]), **match(string, regex)**

Beispiel

Siehe func_regex.qdf

5.3.58 Right(string, count)

Kategorie

String-Funktion

Kurzbeschreibung Gibt die ersten **count** Zeichen, von rechts her gezählt, des Strings **string** zurück.

Argumente

string
String oder Feldname eines Textfelds

count
Anzahl zurückzugebender Zeichen; positive ganze Zahl

Resultat String oder Leerer Wert (wenn **string** weniger Zeichen als **count** aufweist)

Erklärung Gibt die ersten **count** Zeichen, von rechts her gezählt, des Strings **string** zurück. Bei Zählung von links her entspricht dies den letzten **count** Zeichen.

Diese Funktion gibt nur dann einen gültigen Wert zurück, wenn **string** mindestens **count** Zeichen lang ist. Wenn **string** kürzer ist, wird der Leere Wert zurückgegeben.

Siehe auch **Left(string, count)**, **SubStr(string, start[, count])** und übrige Text-Funktionen

Beispiel Siehe func_string1.qdf

5.3.59 Round(x)

Kategorie Mathematik-Funktion

Kurzbeschreibung Rundet den Eingabewert auf die nächste ganze Zahl.

Argumente

x
numerisch, beliebiger Wert

Resultat numerisch

Erklärung Rundet den Eingabewert auf die nächste ganze Zahl auf oder ab. Wenn die Nachkommastellen von x kleiner als x.5 sind, wird abgerundet, wenn sie grösser oder gleich x.5 sind, wird aufgerundet.

Soll auf den Wert genauigkeit gerundet werden, wird

```
= genauigkeit * Round(x/genauigkeit)
```

benutzt

Siehe auch **Ceil(x)**, **Floor(x)**

Beispiel Siehe func_math1.qdf

5.3.60 setFieldPrintable(true|false, field_name)

Kategorie Status-Funktion

Kurzbeschreibung Setzt die Eigenschaft **druckbar** des betreffenden Felds.

Argumente

true|false
Statuswert; **true** oder **false**

field_name
Feldname

Resultat Feldeigenschaft wird gesetzt

Erklärung Setzt die Eigenschaft **druckbar** des Felds **field_name** auf **true** oder **false**. Dies entspricht dem Ein- oder Ausschalten des Kontrollkästchens **druckbar** im Eigenschaften-Dialog des betreffenden Felds.

Siehe auch **setFieldVisible(true|false, field_name)**, **setFieldWritable(true|false, field_name)**

Beispiel Siehe func_status.qdf

5.3.61 setFieldVisible(true|false, field_name)

Kategorie Status-Funktion

Kurzbeschreibung Setzt die Eigenschaft **sichtbar** des betreffenden Felds.

Argumente

true|false
Statuswert; **true** oder **false**

field_name
Feldname

Resultat Feldeigenschaft wird gesetzt

Erklärung Setzt die Eigenschaft **sichtbar** des Felds **field_name** auf **true** oder **false**. Dies entspricht dem Ein- oder Ausschalten des Kontrollkästchens **sichtbar** im Eigenschaften-Dialog des betreffenden Felds.

Siehe auch **setFieldPrintable(true|false, field_name)**, **setFieldWritable(true|false, field_name)**

Beispiel Siehe func_status.qdf

5.3.62 setFieldWritable(true|false, field_name)

Kategorie Status-Funktion

Kurzbeschreibung Setzt sinngemäss die Eigenschaft **schreibgeschützt** des betreffenden Felds.

Argumente

true|false
Statuswert; **true** oder **false**

field_name
Feldname

Resultat

Feldeigenschaft wird gesetzt

Erklärung

Setzt die Eigenschaft **schreibgeschützt** des Felds **field_name** auf **true** oder **false**. Dies entspricht dem Aus- oder Einschalten des Kontrollkästchens **schreibgeschützt** im Eigenschaften-Dialog des betreffenden Felds.

Diese Funktion ist invers zum Ein- und Ausschalten des Kontrollkästchens **schreibgeschützt**. Um dieses einzuschalten muss in dieser Funktion der Status **false** gesetzt werden und umgekehrt.

Siehe auch

setFieldPrintable(true|false, field_name), **setFieldVisible(true|false, field_name)**

Beispiel

Siehe func_status.qdf

5.3.63 setList(ComboBox, str_items, str_values)

Kategorie

Spezial-Funktion

Kurzbeschreibung

Konfiguriert die Auswahlliste für das Auswahllistenfeld **ComboBox**.

Argumente

ComboBox
Name eines Auswahllistenfelds

str_items
String mit der Liste der Elemente der Auswahlliste, umrahmt mit Anführungszeichen und getrennt mit "**|**"

str_values
String mit der Liste der Werte der Auswahlliste, umrahmt mit Anführungszeichen und getrennt mit "**|**"

Resultat

Auswahlliste wird konfiguriert

Erklärung

Diese Funktion dient zum Konfigurieren der Auswahlliste für das Auswahllistenfeld **ComboBox**. Das Anwenden dieser Funktion entspricht dem Ausfüllen der Tabelle im Konfigurationsfenster der Auswahlliste (siehe Abschnitt 4.4.4, Fig. 4-99).

Im String **str_items** werden die Einträge der linken Spalte (Elemente) der Tabelle festgelegt. Die Elemente werden mit dem Zeichen **|** voneinander getrennt.

Im String **str_values** werden die Einträge der rechten Spalte (Werte) der Tabelle festgelegt. Die Elemente werden mit dem Zeichen **|** voneinander getrennt. Wenn dieser String fehlt oder leer ist, sind für die Auswahlliste keine Werte definiert (und es werden bei der Auswahl die Element-Bezeichnungen zurückgegeben.

Es muss darauf geachtet werden, dass die einzelnen Elemente in **str_items** und in **str_values** sich entsprechen. Wenn das nicht der Fall ist, ergeben sich Fehlzuweisungen zwischen Elementbezeichnung und Wert.

Mit dieser Funktion ist es daher möglich, Auswahllisten dynamisch zusammenzustellen. Es ist denkbar, beispielsweise Tabelleninhalte zur Weiterverwendung in eine Auswahlliste zu übertragen.

Siehe auch

getSelectedIndex(ComboBox)

Beispiel

Siehe func_combobox.qdf

5.3.64 setPagePrintable(true|false, page_number [,

page_number[,...]])

Kategorie

Status-Funktion

Kurzbeschreibung

Setzt die Eigenschaft **druckbar** der betreffenden Seite(n).

Argumente

true|false
Statuswert; **true** oder **false**

page_number
Seitennummer oder durch Komma getrennte Gruppe von Seitennummern

Resultat

Seiteneigenschaft wird gesetzt

Erklärung

Setzt die Eigenschaft **druckbar** von einer oder mehrerer Seiten **page_number** auf **true** oder **false**. Dies entspricht dem Ein- oder Ausschalten des Kontrollkästchens **druckbar** im Eigenschaften-Dialog der betreffenden Seite (siehe Abschnitt 4.3.4.2, Fig. 4-18).

Siehe auch

setPageVisible(true|false, page_number [, page_number[, ...]])

Beispiel

Siehe func_status.qdf

5.3.65 setPageVisible(true|false, page_number [, page_number[, ...]])

Kategorie

Status-Funktion

Kurzbeschreibung

Setzt die Eigenschaft **sichtbar** der betreffenden Seite(n).

Argumente

true|false
Statuswert; **true** oder **false**

page_number
Seitennummer oder durch Komma getrennte Gruppe von Seitennummern

Resultat Seiteneigenschaft wird gesetzt

Erklärung Setzt die Eigenschaft **sichtbar** von einer oder mehrerer Seiten **page_number** auf **true** oder **false**. Dies entspricht dem Ein- oder Ausschalten des Kontrollkästchens **sichtbar** im Eigenschaften-Dialog der betreffenden Seite (siehe Abschnitt 4.3.4.2, Fig. 4-18).

Siehe auch **setPagePrintable(true|false, page_number [, page_number[,…]])**

Beispiel Siehe func_status.qdf

5.3.66 setPrintable(category_list, true|false)

Kategorie Status-Funktion

Kurzbeschreibung Setzt die Eigenschaft **druckbar** der betreffenden Kategorie(n).

Argumente

category_list
String mit durch Komma getrennter Liste von Kategorien

true|false
Statuswert; **true** oder **false**

Resultat Feldeigenschaft für Felder der betreffenden Kategorie wird gesetzt

Erklärung Setzt die Eigenschaft **druckbar** der in einer oder mehreren Kategorien gemäss **category_list** enthaltenen Felder auf **true** oder **false**. Dies entspricht dem gleichzeitigen Ein- oder Ausschalten des Kontrollkästchens **druckbar** im Eigenschaften-Dialog der betroffenen Felder.

Siehe auch **setVisible(category_list, true|false)**, **setWritable(category_list, true|false)**

Beispiel Siehe func_status.qdf

5.3.67 setSelectedIndex(ComboBox, index)

Kategorie Spezial-Funktion

Kurzbeschreibung Setzt die aktuelle Auswahl des Listenfelds **ComboBox** auf den Wert mit Index **index**.

Argumente

ComboBox
Name eines Auswahllistenfelds

index
Index der zu setzenden Auswahl aus der Auswahlliste; ganze Zahl zwischen 1 und der Anzahl Einträge des Auswahllistenfelds

Resultat Der Eintrag mit Index **index** wird zur aktuellen Auswahl

Erklärung Setzt die aktuelle Auswahl des Listenfelds ComboBox auf den Eintrag mit Index **index**.

Siehe auch **getSelectedIndex(ComboBox)**, **setList(ComboBox, str_items, str_values)**

Beispiel Siehe func_combobox.qdf

5.3.68 setVisible(category_list, true|false)

Kategorie Status-Funktion

Kurzbeschreibung

Setzt die Eigenschaft **sichtbar** der betreffenden Kategorie(n).

Argumente

category_list
String mit durch Komma getrennter Liste von Kategorien

true|false
Statuswert; **true** oder **false**

Resultat

Feldeigenschaft für Felder der betreffenden Kategorie wird gesetzt

Erklärung

Setzt die Eigenschaft **sichtbar** der in einer oder mehreren Kategorien gemäss **category_list** enthaltenen Felder auf **true** oder **false**. Dies entspricht dem gleichzeitigen Ein- oder Ausschalten des Kontrollkästchens **sichtbar** im Eigenschaften-Dialog der betroffenen Felder.

Siehe auch

setPrintable(category_list, true|false), **setWritable(category_list, true|false)**

Beispiel

Siehe func_status.qdf

5.3.69 setWritable(category_list, true|false)

Kategorie

Status-Funktion

Kurzbeschreibung

Setzt sinngemäss die Eigenschaft **schreibgeschützt** der betreffenden Kategorie(n).

Argumente

category_list
String mit durch Komma getrennter Liste von Kategorien

true|false
Statuswert; **true** oder **false**

Resultat
Feldeigenschaft für Felder der betreffenden Kategorie wird gesetzt

Erklärung
Setzt die Eigenschaft **schreibgeschützt** der in einer oder mehreren Kategorien gemäss **category_list** enthaltenen Felder auf **true** oder **false**. Dies entspricht dem gleichzeitigen Ein- oder Ausschalten des Kontrollkästchens **schreibgeschützt** im Eigenschaften-Dialog der betroffenen Felder.

Diese Funktion ist invers zum Ein- und Ausschalten des Kontrollkästchens **schreibgeschützt**. Um dieses einzuschalten muss in dieser Funktion der Status **false** gesetzt werden und umgekehrt.

Siehe auch
setPrintable(category_list, true|false), **setVisible(category_list, true|false)**

Beispiel
Siehe func_status.qdf

5.3.70 Sin(x)

Kategorie
Mathematik-Funktion

Kurzbeschreibung
Berechnet den Sinus eines in Grad definierten Winkels.

Argumente
x
numerisch, beliebiger Wert, die Eingabewerte wiederholen sich in Zyklen von 360°

Resultat
numerisch, liegt zwischen **-1** und **+1**

Erklärung
Gibt den Sinus des Eingabewerts zurück. Der Eingabewert wird in Grad genommen. Der volle Kreis entspricht 360°.

Siehe auch
Asin(x), **Acos(x)**, **Atan(x)**, **Cos(x)**, **Tan(x)**, übrige Mathematik-Funktionen

Beispiel Siehe func_math2.qdf

5.3.71 Sqrt(x)

Kategorie Mathematik-Funktion

Kurzbeschreibung Berechnet die Quadratwurzel des Eingabewerts.

Argumente **x**
numerisch, Wert grösser 0

Resultat numerisch

Erklärung Gibt die Quadratwurzel des Eingabewerts zurück. Wenn der Eingabewert negativ ist, ist die Quadratwurzel nicht definiert, und es wird eine Fehlermarke angezeigt.

Siehe auch Operator **x ^ y**

Beispiel Siehe func_math2.qdf

5.3.72 Str(x)

Kategorie String-Funktion

Kurzbeschreibung Wandelt den Eingabewert in einen String um.

Argumente **x**
String, Zahl, Datum, boolean

Resultat String

Erklärung Gibt die String-Darstellung des Eingabewerts zurück.

Siehe auch **Len(x)**, **SubStr(string, start[, count])**

Beispiel Siehe func_string1.qdf

5.3.73 StrTotal(table, column)

Kategorie Tabellen-Funktion

Kurzbeschreibung Fügt die Tabellenwerte in Spalte **column** zu einem String zusammen.

Argumente

table
Name der Tabelle

column
Nummer oder Feldname der Spalte, über welcher die Feldwerte zusammengestellt werden sollen

Resultat String

Erklärung Fügt die Feldwerte der Spalte **column** der Tabelle **table** zu einem String zusammen. Dabei wird "**|**" als Trennzeichen benutzt.

Siehe auch **TblValues(table, tbl_start, tbl_end, row_start, row_end, delimiter, column [, column[, ...]])**, **Total(table, column)**, **TimeTotal(table, column)**

Beispiel Siehe func_table1.qdf

5.3.74 SubArray(byte_array, start [, count])

Kategorie String-Funktion

Kurzbeschreibung
Bildet eine Sub-Folge einer Bytefolge **byte_array**, beginnend mit Byte Nummer **start** und der Anzahl **count** Bytes.

Argumente
byte_array
Bytefolge, normalerweise erzeugt mit **Compress(x)** oder **getBytes(string)**

start
Ganze Zahl, zwischen **0** und der Länge der Bytefolge **byte_array**

count (optional)
Ganze Zahl, zwischen 1 und der Länge der Bytefolge **byte_array**; wenn der Wert fehlt, wird bis zum Ende der Bytefolge extrahiert

Resultat
Bytefolge

Erklärung
Diese Funktion ist das selbe wie **SubStr(string, start[, count])** zu einem String, aber für Bytefolgen.

Sie erzeugt eine Sub-Bytefolge der Bytefolge **byte_array**, beginnend ab Zeichenposition **start** und mit der Länge **count**. Wenn **count** nicht definiert ist, wird bis zum Ende der Bytefolge extrahiert. Die Startposition beginnt mit **0**.

Siehe auch
getBytes(string), **Compress(x)**

Beispiel
Siehe func_string2.qdf und func_barcode.qdf

5.3.75 SubStr(string, start[, count])

Kategorie
String-Funktion

Kurzbeschreibung
Bildet einen Teilstring ab Zeichenposition **start** mit der Länge **count**.

Argumente

string
String, Direkteingabe oder Resultat einer Formel

start
Zeichenposition, ab welcher der Teilstring gebildet werden soll; ganze Zahl zwischen **0** und der Länge von **string**

count (optional)
Anzahl Zeichen des Teilstrings; ganze Zahl zwischen **1** und der Länge von **string**

Resultat

String

Erklärung

Erzeugt einen Teilstring von string mit Länge **count**, beginnend ab Position **start**.

Die Numerierung der Positionen beginnt mit **0**. Mit **0** als Startwert entspricht diese Funktion eigentlich der Funktion **Left(string, count)**.

Wenn kein Wert für **count** spezifiziert wird, wird der Teilstring bis zum Ende von **string** genommen (was dann eigentlich mit der Funktion **Right(string, count)** übereinstimmt).

Siehe auch

Right(string, count), **Left(string, count)**, **Str(x)**

Beispiel

Siehe func_string1.qdf

5.3.76 Tan(x)

Kategorie

Mathematik-Funktion

Kurzbeschreibung

Berechnet den Tangens eines in Grad definierten Winkels.

Argumente

x
numerisch, beliebiger Wert, die Eingabewerte wiederholen sich in Zyklen von 180°

Resultat

numerisch

Erklärung

Gibt den Tangens des Eingabewerts zurück. Der Eingabewert wird in Grad genommen. Der volle Kreis entspricht 360°.

Siehe auch

Atan(x), **Asin(x)**, **Acos(x)**, **Cos(x)**, **Sin(x)**, übrige Mathematik-Funktionen

Beispiel

Siehe func_math2.qdf

5.3.77 TblValues(table, tbl_start, tbl_end, row_start, row_end, delimiter, column [, column[, …]])

Kategorie

Tabellen-Funktion

Kurzbeschreibung

Erzeugt einen String mit zeilenweise zusammengefügten Tabellenwerten und spezifizierten Delimitern.

Argumente

table
Bezeichnung der Tabelle

tbl_start
String zur Kennung des Starts der Tabelle

tbl_end
String zur Kennung des Endes der Tabelle

row_start
String zur Kennung des Starts einer Tabellenzeile

row_end
String zur Kennung des Endes einer Tabellenzeile

delimiter
String zur Trennung der Feldwerte innerhalb einer Tabellenzeile

column (wiederholbar)
Spaltennummer oder Spaltenbezeichnung der Tabelle

Resultat

String mit einer durch die Argumente definierten Struktur

Erklärung

Diese Funktion dient zum Export von Tabelleninhalten, sei es für die Datenübertragung, sei es für die Aufbereitung für einen 2D-Barcode. Die Möglichkeit, Strings zur Kennung von Tabellen-Start und -Ende sowie Zeilen-Start und -Ende zu definieren erlaubt es, den Tabellenexport auch in XML-Form vorzunehmen.

Die Kennungen und **delimiter** können aus einem oder mehreren Zeichen bestehen. Ebenso kann ein leerer String oder überhaupt kein Zeichen definiert werden. Ein Anführungszeichen sowie ein Backslash selbst muss mit Backslash (****) geschützt werden. Für eine neue Zeile (zum Beispiel für row_end) muss **\n** eingegeben werden.

Die Spalten können in freier Reihenfolge angeordnet werden. Sogar ein Wiederholen von Spaltenwerten ist möglich.

Diese Funktion gibt Spaltensummen nicht aus. Diese müssten separat hinzugefügt werden.

Siehe auch

Total(table, column), **StrTotal(table, column)**, **TimeTotal(table, column)**

Beispiel

Siehe func_table1.qdf

5.3.78 Time(hour, minute[, second])

Kategorie

Datum-Funktion

Kurzbeschreibung

Erzeugt ein Uhrzeitobjekt aus den Argumenten.

Argumente

hour
Ganze positive Zahl, entsprechend der Stundenzahl

minute
Ganze positive Zahl, entsprechend der Minutenzahl

second (optional)
Ganze positive Zahl, entsprechend der Sekundenzahl

Resultat

Uhrzeit

Erklärung

Erzeugt eine Uhrzeit aus den drei Argumenten. Die Eingabe für **hour** sollte zwischen 0 und 24 liegen.

Der Wert für **minute** sollte zwischen 0 und 59 liegen. Grössere Werte werden umgerechnet und die Stundenzahl entsprechend erhöht.

Der Wert für **second** ist optional und sollte zwischen 1 und 59 liegen. Grössere Werte werden umgerechnet und entsprechend der Minute und Stunde zugeschlagen.

Die Eingaben sollten ganze Zahlen sein, wobei trotzdem eingegebene Nachkommastellen ignoriert werden.

Siehe auch

getHour(time_value), **getMinute(time_value)**, sowie übrige Datum-Funktionen

Beispiel

Siehe func_time.qdf

5.3.79 TimeTotal(table, column)

Kategorie

Tabellen-Funktion

Kurzbeschreibung

Summiert die Werte der Spalte **column** zu einer Uhrzeit.

Argumente

table
Name der Tabelle

column
Nummer oder Feldname der Spalte, über welcher die Feldwerte summiert werden sollen

Resultat

Uhrzeit

Erklärung

Summiert die Feldwerte der Spalte **column** als Uhrzeit und gibt das Resultat zurück. Es muss daher darauf geachtet werden, dass die Spalte **column** tatsächlich als Uhrzeit formatiert ist.

Siehe auch

Total(table, column), **StrTotal(table, column)**

Beispiel

Siehe func_table2.qdf

5.3.80 Today()

Kategorie

Datums-Funktion

Kurzbeschreibung

Gibt das aktuelle Datum zurück.

Argumente

Diese Funktion hat keine Argumente.

Resultat

Datum

Erklärung

Gibt das aktuelle Datum zurück. Um dieses darzustellen ist ein als Datum formatiertes Datumsfeld notwendig.

Siehe auch

getDay(date_value), **getMonth(date_value)**, **getYear(date_value)** und die übrigen Datum-orientierten Datums-Funktionen

Beispiel

Siehe func_date.qdf

5.3.81 Total(table, column)

Kategorie

Tabellen-Funktion

Kurzbeschreibung

Summiert die Werte der Spalte **column** zu Summe.

Argumente

table
Name der Tabelle

column
Nummer oder Feldname der Spalte, über welcher die Feldwerte summiert werden sollen

Resultat

Zahl

Erklärung

Summiert die Feldwerte der Spalte **column** zu einer Summe und gibt das Resultat zurück. Es muss daher darauf geachtet werden, dass die Spalte **column** tatsächlich ein Zahlenfeld ist.

Siehe auch

TimeTotal(table, column), **StrTotal(table, column)**

Beispiel

Siehe func_table1.qdf

5.3.82 TrimStr(string)

Kategorie String-Funktion

Kurzbeschreibung Entfernt vor- und nachlaufende Leerzeichen von einem String.

Argumente **string**
String; kann vor und nachlaufende Leerzeichen haben

Resultat String

Erklärung Entfernt allfällige vor- und nachlaufende Leerzeichen beim Eingabe-String und gibt den "beschnittenen" String zurück. Wenn der String keine vor- oder nachlaufenden Leerzeichen besitzt ist das Resultat mit dem Eingabestring identisch. Diese Funktion ist insbesonder beim Zusammenstellen von Daten für 2D-Barcodes nützlich.

Siehe auch **SubStr(string, start[, count])**, **InStr(string, pattern [, start_position])**

Beispiel Siehe func_string2.qdf

5.3.83 Trunc(x)

Kategorie Mathematik-Funktion

Kurzbeschreibung Rundet den Absolutwert des Eingabewerts auf die nächstkleinere ganze Zahl ab.

Argumente **x**
numerisch, beliebiger Wert

Resultat numerisch

Erklärung

Gibt, ohne auf das Vorzeichen zu achten, den auf die nächstkleinere ganze Zahl abgerundeten Eingabewert zurück. Diese Funktion entspricht der Gleichung

$$\text{Trunc}(x) = \frac{x}{\text{Abs}(x)} \times \text{Floor}(x)$$

Diese Funktion ist daher für positive Zahlen identisch mit **Floor(x)**; für negative Zahlen weicht der Zahlenwert gegenüber von **Floor(x)** um 1 ab.

Siehe auch

Floor(x), **Round(x)**, **Ceil(x)**, **Cover(x)**

Beispiel

Siehe func_math1.qdf

5.4 Regular Expressions

Regular Expressions (dieser Begriff wird auf diese Art und Weise auch in deutscher Sprache verwendet) sind ein äusserst mächtiges Werkzeug zum Auswerten von Zeichenketten. Sie erlauben das Finden von Mustern (pattern), welche bestimmten Regeln folgen. Sie erlauben zudem das Ersetzen der Fundstellen durch andere Zeichenketten, was einer äusserst mächtigen "Suchen und Ersetzen"-Funktion entspricht.

Die nachstehende Tabelle ist von der Programm-Entwicklungs-Dokumentation abgeleitet und fasst die wichtigsten Syntaxelemente zum Aufbau von Regular Expressions zusammen. Für eine weiter gehende Betrachtung von Regular Expressions wird auf die Literatur verwiesen. Als Standardwerk gilt das Buch von Jeffrey Friedl:

Mastering Regular Expressions, 2nd Edition, Jeffrey E. F. Friedl, O'Reilly and Associates, 2002 (Amazon Link).

5.4.1 Übersichtstabelle

Syntaxelement	Fundstelle
	Zeichen
x	Das Zeichen *x*
\\	Das Backslash \ Zeichen
\0*n*	Das Zeichen mit Oktal-Wert 0*n* (0 <= *n* <= 7)
\0*nn*	Das Zeichen mit Oktal-Wert 0*nn* (0 <= *n* <= 7)
\0*mnn*	Das Zeichen mit Oktal-Wert 0*mnn* (0 <= *m* <= 3, 0 <= *n* <= 7)
\x*hh*	Das Zeichen mit Hexadezimal-wert 0x*hh*
\u*hhhh*	Das Zeichen mit Unicode (Hexadezimal-)Wert 0x*hhhh*
\t	Das Tab-Zeichen ('\u0009')
\n	Das Line Feed-Zeichen ('\u000A')
\r	Das Carriage Return-Zeichen ('\u000D')
\f	Das Form Feed-Zeichen ('\u000C')
\a	Das Glocken- (Bell) Zeichen ('\u0007')
\e	Das Escape-Zeichen ('\u001B')
\c*x*	Das Control-Zeichen, entsprechen *x*
	Zeichengruppen

[abc]	a, b, oder c (einfache Gruppe)
[^abc]	Beliebiges Zeichen, ausser a, b, oder c (Negation)
[a-zA-Z]	a bis z oder A bis Z (Bereich)
[a-d[m-p]]	a bis d, oder m bis p: [a-dm-p] (Vereinigung)
[a-z&&[def]]	d, e, oder f (Schnittmenge)
[a-z&&[^bc]]	a bis z, ausser b und c: [ad-z] (Subtraktion)
[a-z&&[^m-p]]	a bis z, jedoch nicht m bis p: [a-lq-z] (Subtraktion)

Vordefinierte Zeichengruppen

.	Beliebiges Zeichen (mit oder ohne Zeilenende)
\d	Eine Ziffer: [0-9]
\D	Nicht eine Ziffer: [^0-9]
\s	Ein Weissraum-Zeichen: [\t\n\x0B\f\r]
\S	Nicht ein Weissraum-Zeichen: [^\s]
\w	Ein Wort-Zeichen: [a-zA-Z_0-9]
\W	Nicht ein Wort-Zeichen: [^\w]

POSIX Zeichengruppen (nur für US-ASCII)

\p{Lower}	Ein Kleinbuchstabe: [a-z]
\p{Upper}	Ein Grossbuchstabe: [A-Z]
\p{ASCII}	Beliebiger ASCII-Wert: [\x00-\x7F]
\p{Alpha}	Ein Buchstabe: [\p{Lower}\p{Upper}]
\p{Digit}	Eine Ziffer: [0-9]
\p{Alnum}	Ein alphanumerisches Zeichen: [\p{Alpha}\p{Digit}]
\p{Punct}	Ein Satzzeichen !"#$%&'()*+,-./:;<=>?@[\]^_`{|}~
\p{Graph}	Ein sichtbares Zeichen [\p{Alnum}\p{Punct}]
\p{Print}	Ein druckbares Zeichen: [\p{Graph}]
\p{Blank}	Leerzeichen oder Tab-Zeichen: [\t]
\p{Cntrl}	Ein Control-Zeichen [\x00-\x1F\x7F]
\p{XDigit}	Ein Hexadezimal-Zeichen: [0-9a-fA-F]
\p{Space}	Ein Weissraum-Zeichen [\t\n\x0B\f\r]

Zeichengruppen für Unicode-Blocks und -Kategorien

\p{InGreek}	Ein Zeichen des Griechisch-Block (einfacher Block)
\p{Lu}	Ein Grossbuchstabe (einfache Kategorie)
\p{Sc}	Ein Währungssymbol
\P{InGreek}	Ein Zeichen ausserhalb des Griechisch-Blocks (Negation)
[\p{L}&&[^\p{Lu}]]	Beliebiges Zeichen ausser Grossbuchstaben (Subtraktion)

Abgrenzungen

^	Beginn einer Zeile
$	Ende einer Zeile
\b	Wort-Abgrenzung
\B	Keine Wort-Abgrenzung
\A	Beginn der Eingabe
\G	Ende der vorherigen Fundstelle

\Z	Ende der Eingabe ausser dem letzten Abschlusszeichen, falls vorhanden
\z	Ende der Eingabe
	Gierige Vervielfacher
X?	*X*, ein- oder kein-mal
*X**	*X*, null oder beliebige Anzahl mal
X+	*X*, ein- oder mehrmals
X{ *n*}	*X*, genau *n* mal
X{ *n*,}	*X*, mindestens *n* mal
X{ *n*,*m*}	*X*, mindestens *n* aber höchstens *m* mal
	Zurückhaltende Vervielfacher
X??	*X*, ein- oder kein-mal
*X**?	*X*, null oder beliebige Anzahl mal
X+?	*X*, ein- oder mehrmals
X{ *n*}?	*X*, genau *n* mal
X{ *n*,}?	*X*, mindestens *n* mal
X{ *n*,*m*}?	*X*, mindestens *n* aber höchstens *m* mal
	Besitzergreifende Vervielfacher
X?+	*X*, ein- oder kein-mal
*X**+	*X*, null oder beliebige Anzahl mal
X++	*X*, ein- oder mehrmals
X{ *n*}+	*X*, genau *n* mal
X{ *n*,}+	*X*, mindestens *n* mal
X{ *n*,*m*}+	*X*, mindestens *n* aber höchstens *m* mal
	Logische Operatoren
XY	*X* gefolgt von *Y*
X|*Y*	Entweder *X* oder *Y*
(*X*)	*X*, als eine gespeicherte Gruppe
	Rückwärts-Referenzen
n	Fundstelle der *n*-ten gespeicherten Gruppe

Praxistip: *Das Backslash-Zeichen ("\") dient zur Anzeige von geschützten Ausdrücken, wie in der obigen Tabelle gezeigt. wie auch zur Anzeige von Zeichen, welche ansonsten als Steuerzeichen betrachtet würden (z.B. der Backslash selbst).*

6 Snapform-Dateiformate

6.1 Formular, komprimiert (QDF/S)

Dateiendung

.qdf

Erklärung

Kompaktes Snapform-Format. Der gesamte Inhalt ist komprimiert und verschlüsselt. Benötigt besondere Tools zum Auslesen der Daten, einschliesslich der Metadaten.

Formatbeschreibung

Rein binäres Format; Auslesen der Daten nur via API (Format-Beschreibung siehe API-Dokumentation).

6.2 Formular, offen (QDF/A)

Dateiendung

.qdf

Erklärung

Offenes Snapform-Format. Nur die Formularbeschreibung ist komprimiert und verschlüsselt; Daten und Metadaten sind in einer offenen XML-Form. Zum Lesen und Schreiben der Daten werden keine besondere Tools benötigt.

Formatbeschreibung

Das QDF-Format ist im Snapform-Schema beschrieben. Das Schema ist unter **http://www.snapform.com/open_format/snapform.xsd** zugänglich. Es ist zudem in Abschnitt 6.9 aufgeführt.

Der nachstehende Code zeigt ein auf die wesentlichen Elemente reduziertes Beispiel.

```
<?xml version="1.0" encoding="UTF-8"?>
<QDFA version="1.5" xmlns:xsi="http://www.w3.org/2001/
XMLSchema-instance" xsi:noNamespaceSchemaLocation="http:/
/www.snapform.com/open_format/snapform.xsd"
xsi:rdf="http://www.snapform.com/open_format/rdf.xsd">
    <Metadata>
```

```
        <rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-
rdf-syntax-ns#">
        <rdf:Description rdf:about="Dublin Core"
xmlns:dc="http://purl.org/dc/elements/1.1/">
        <dc:Title>REQUEST FOR SERVICES</dc:Title>
        <dc:Date>08-2005</dc:Date>
        <dc:Identifier>NRC FORM 160</dc:Identifier>
        </rdf:Description>
        </rdf:RDF>
    </Metadata>
    <Data>
        <snapform>
        <id>AF14A833-274C-C57A-0237-015E0D8A3F54</id>
        <values>
        <value name="contractor" type="4">false</value>
        <value name="date1" type="5"></value>
        <value name="form_no" type="3"></value>
        <value name="hq" type="4">false</value>
        <value name="licensee" type="4">false</value>
        </values>
    </snapform>
    </Data>
    <Form><![CDATA[QlpoOTFBWSZTWREBGEgAA07//////////////
/////////////
+N71AAAASKCgAAoFUAqm9fW4vVbs+3p5eOAAKAAjxMqp0dAo

...

DHALAPr8xrPEfcjiG+IZwcklbOMHyAwdKMAd1OUGhvD2o4B60dUN6wwV1
A1sBg5EccH/4u5IpwoSAiAjCQA==]]>
    </Form>
</QDFA>
```

Nachstehend sind die Elemente kommentiert

```
<?xml version="1.0" encoding="UTF-8"?>
```

XML-Header.

```
<QDFA version="1.5" xmlns:xsi="http://www.w3.org/2001/
XMLSchema-instance" xsi:noNamespaceSchemaLocation="http:/
/www.snapform.com/open_format/snapform.xsd"
xsi:rdf="http://www.snapform.com/open_format/rdf.xsd">
```

QDFA ist das Hauptelement eines QDF/A-Formulars

```
    <Metadata>
```

Das **Metadata**-Element umfasst die im Formulareigenschaften, Metadaten-Fenster erfassten Metadaten (siehe Abschnitt 4.8.5).

```
        <rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-
rdf-syntax-ns#">
        <rdf:Description rdf:about="Dublin Core"
xmlns:dc="http://purl.org/dc/elements/1.1/">
```

Es werden die Metadaten gemäss Dublin Core-Spezifikation verwendet.

```
        <dc:Title>REQUEST FOR SERVICES</dc:Title>
        <dc:Date>08-2005</dc:Date>
        <dc:Identifier>NRC FORM 160</dc:Identifier>
        </rdf:Description>
        </rdf:RDF>
    </Metadata>


    <Data>
```

Der **Data**-Block umfasst die Nutzdaten.

```
        <snapform>
```

Das Element **snapform** besagt, dass es sich tatsächlich um Snapform-Daten handelt.

```
        <id>AF14A833-274C-C57A-0237-015E0D8A3F54</id>
```

Das Element **id** ist die Formular-ID (siehe Abschnitt 4.8.2).

```
        <values>
```

Im Element **values** sind die eigentlichen Feldwerte (Element **value**) zusammengefasst.

```
        <value name="contractor" type="4">false</value>
        <value name="date1" type="5"></value>
        <value name="date2" type="5"></value>
        <value name="form_no" type="3"></value>
        <value name="licensee" type="4">false</value>
        <value name="mailstop" type="3"></value>
        </values>
    </snapform>
    </Data>
    <Form><![CDATA[QlpoOTFBWSZTWREBGEgAA07///////////////
/////////////
+N71AAAASKCgAAoFUAqm9fW4vVbs+3p5eOAAKAAjxMqp0dAo

...

DHALAPr8xrPEfcjiG+IZwcklbOMHyAwdKMAd1OUGhvD2o4B60dUN6wwV1
A1sBg5EccH/4u5IpwoSAiAjCQA==]]>
    </Form>
```

Im Element **Form** ist ein Blob mit der codierten Formulardefinition enthalten, welche von Snapform Designer und Snapform Viewer interpretiert wird.

```
</QDFA>
```

6.3 Formular-Template, komprimiert (QDF/S)

Dateiendung

.qdft

Erklärung

Kompaktes Snapform-Template-Format. Der gesamte Inhalt ist komprimiert und verschlüsselt. Benötigt besondere Tools zum Auslesen der Daten, einschliesslich der Metadaten.

Das Template-Format forciert im Snapform Viewer ein **Speichern unter…** mit einem anderen Namen.

Formatbeschreibung

Rein binäres Format; Auslesen der Daten nur via API (Format-Beschreibung siehe API-Dokumentation).

6.4 Formular-Template, offen (QDF/A)

Dateiendung

.qdft

Erklärung

Offenes Snapform-Format. Nur die Formularbeschreibung ist komprimiert und verschlüsselt; Daten und Metadaten sind in einer offenen XML-Form. Zum Lesen und Schreiben der Daten werden keine besondere Tools benötigt.

Das Template-Format forciert im Snapform Viewer ein **Speichern unter…** unter einem anderen Namen.

Formatbeschreibung

Die Formatbeschreibung ist mit dem “normalen” PDF/A-Format identisch (siehe Abschnitt 6.2).

6.5 Hintergrundebene

Dateiendung **.hdt**

Erklärung Format der Hintergrundebene, welche mit Hilfe des Snapform PDF Converters aus einem PDF-Dokument erzeugt worden ist.

Beim Öffnen einer Hintergrundebene-Datei in Snapform Designer wird ein neues Dokument geschaffen.

Formatbeschreibung Rein binäres Format; Auslesen der Daten nur via API (Beschreibung siehe API-Dokumentation).

6.6 Feldebene

Dateiendung **.hdo**

Erklärung Format der Feldebene, welche aus dem Snapform Designer exportiert werden kann.

Beim Öffnen einer Feldebene-Datei in Snapform Designer wird ein neues Dokument geschaffen.

Formatbeschreibung Rein binäres Format; Auslesen der Daten nur via API (Beschreibung siehe API-Dokumentation).

6.7 Daten, binär

Dateiendung **.qdd**

Erklärung Format der binären Expoft- und Import-Datei, welche in Snapform Viewer benutzt wird.

Dieses Format ist komprimiert und verschlüsselt und ist daher sehr kompakt. Es benötigt besondere Tools zum Auslesen und Einschreiben der Daten.

Formatbeschreibung

Rein binäres Format; Auslesen der Daten nur via API (Beschreibung siehe API-Dokumentation).

6.8 Daten, offen

Dateiendung

.xml

Erklärung

Format der offenen Expoft- und Import-Datei, welche in Snapform Viewer benutzt wird.

Dieses Format ist offen und wie eine normale XML-Datei behandelbar. Zum Auslesen und Einschreiben der Daten können handelsübliche Tools eingesetzt werden.

Formatbeschreibung

Dieses Format ist eine Teilmenge des QDF/A-Formats, und benutzt nur das Data-Element des Snapform-Schemas (siehe Abschnitt 6.2).

6.9 Das Snapform-Schema

Das Snapform-Schema ist unter **http://www.snapform.com/open_format/snapform.xsd** zugänglich.

Das Schema ist nachstehend aufgeführt. Es empfiehlt sich jedoch, die Datei herunterzuladen und in einem geeigneten XML-Editor (XMLSpy, oXygen) anzuzeigen. Auf eine weitergehende Interpretation des Schemas wird verzichtet.

```
<?xml version="1.0" encoding="UTF-8"?>
<!-- edited with XMLSPY v5 rel. 4 U (http://www.xmlspy.com)
by Ringler Informatik AG -->
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
elementFormDefault="qualified">
    <xs:element name="id"/>
```

```
    <xs:element name="value">
        <xs:complexType mixed="true">
        <xs:sequence minOccurs="0">
        <xs:element ref="rows"/>
        </xs:sequence>
        <xs:attribute name="name" type="xs:string"
use="required"/>
        <xs:attribute name="type" type="xs:integer"
use="required">
        <xs:annotation>
        <xs:documentation>
        Possible values for attribute 'type' are:
        1  - Integer value
        2  - Double value
        3  - String value
        4  - Boolean value
        5  - Date value
        6  - Table
        7  - Array of bytes
        8  - String value with index (used in dropdown
lists)
        9  - Label value
        10 - Negative double value</xs:documentation>
        </xs:annotation>
        </xs:attribute>
        <xs:attribute name="calculated" type="xs:boolean"
use="optional" default="false"/>
        <xs:attribute name="visible" type="xs:boolean"
use="optional" default="true"/>
        <xs:attribute name="printable" type="xs:boolean"
use="optional" default="true"/>
        <xs:attribute name="readonly" type="xs:boolean"
use="optional" default="false"/>
        </xs:complexType>
    </xs:element>
    <xs:element name="row">
        <xs:complexType>
        <xs:sequence>
        <xs:element ref="value" maxOccurs="unbounded"/>
        </xs:sequence>
        </xs:complexType>
    </xs:element>
    <xs:element name="rows">
        <xs:complexType>
        <xs:sequence>
        <xs:element ref="row" maxOccurs="unbounded"/>
        </xs:sequence>
        </xs:complexType>
    </xs:element>
    <xs:element name="values">
        <xs:complexType>
        <xs:sequence>
        <xs:element ref="value" maxOccurs="unbounded"/>
        </xs:sequence>
        </xs:complexType>
    </xs:element>
    <xs:element name="snapform">
        <xs:complexType>
        <xs:sequence>
        <xs:element ref="id"/>
```

```
        <xs:element ref="values"/>
        </xs:sequence>
        </xs:complexType>
    </xs:element>
    <xs:element name="Keywords">
        <xs:complexType mixed="true">
        <xs:sequence>
        <xs:any processContents="skip"
maxOccurs="unbounded"/>
        </xs:sequence>
        <xs:attribute name="URL" type="xs:string"
use="optional"/>
        </xs:complexType>
    </xs:element>
    <xs:element name="Metadata">
        <xs:complexType>
        <xs:sequence>
        <xs:any processContents="skip"
maxOccurs="unbounded"/>
        </xs:sequence>
        <xs:attribute name="URL" type="xs:string"
use="optional"/>
        </xs:complexType>
    </xs:element>
    <xs:element name="Data">
        <xs:complexType>
        <xs:sequence>
        <xs:element ref="snapform"/>
        </xs:sequence>
        <xs:attribute name="URL" type="xs:string"
use="optional"/>
        </xs:complexType>
    </xs:element>
    <xs:element name="Form">
        <xs:complexType mixed="true">
        <xs:sequence>
        <xs:any processContents="skip"
maxOccurs="unbounded"/>
        </xs:sequence>
        <xs:attribute name="URL" type="xs:string"
use="optional"/>
        </xs:complexType>
    </xs:element>
    <xs:element name="Attachment">
        <xs:complexType mixed="true">
        <xs:sequence>
        <xs:any processContents="skip"
maxOccurs="unbounded"/>
        </xs:sequence>
        <xs:attribute name="Name" type="xs:string"
use="required"/>
        <xs:attribute name="Timestamp" type="xs:string"
use="optional"/>
        </xs:complexType>
    </xs:element>
    <xs:element name="Attachments">
        <xs:complexType>
        <xs:sequence>
        <xs:element ref="Attachment"/>
        </xs:sequence>
```

```
        </xs:complexType>
    </xs:element>
    <xs:element name="QDFA">
        <xs:complexType>
        <xs:all>
        <xs:element ref="Keywords" minOccurs="0"/>
        <xs:element ref="Metadata" minOccurs="0"/>
        <xs:element ref="Data" minOccurs="0"/>
        <xs:element ref="Form" minOccurs="0"/>
        <xs:element ref="Attachments" minOccurs="0"/>
        </xs:all>
        <xs:attribute name="version" type="xs:string"
use="optional"/>
        </xs:complexType>
    </xs:element>
</xs:schema>
```

7 API und Toolkits

Ein API sowie Toolkits stehen zur Verfügung. Ihre Dokumentation ist ausserhalb des Umfangs dieses Handbuchs.

Weitere Informationen sind bei Ringler Informatik AG erhältlich.